DOCUMENT RESUME

ED 148 340 IR 005 441

AUTHOR Williams, Martha E.; MacLaury, Keith D.
TITLE A State-Wide Union Catalog Feasibility Study. Final Report on Project III--FY 1976.
INSTITUTION Illinois Univ., Urbana. Coordinated Science Lab.
SPONS AGENCY Illinois State Library, Springfield.
PUB DATE Oct 77
NOTE 80p.

EDRS PRICE MF-$0.83 HC-$4.67 Plus Postage.
DESCRIPTORS *Cataloging; Computer Storage Devices; *Feasibility Studies; Information Storage; *Library Automation; Library Research; *Library Technical Processes; *Union Catalogs; University Libraries

ABSTRACT

The objective of the project was to develop an inexpensive and effective technique for matching records in multiple fields in order to build a union catalog containing only one representation of a given book. The Information Retrieval Research Lab (IRRL) developed a prototype system called IUCS (IRRL Union Catalog System) by testing various techniques for matching records against large test files of machine-readable records from major institutions that maintain their catalogs in machine-readable form. IUCS involves four major steps: (1) a pre-processing step which converts incoming files into IUCS form; (2) a first pass screen which matches records according to an efficient title-date matching key; (3) a second pass screen which provides more refined matching of the potential duplicates located in the first pass; and (4) a post-processing step which merges the new records into the existing MASTER file to create a union catalog. These procedures were tested by generating a union catalog of three sample files from Ohio College Library Center, Northwestern University, and the University of Chicago. The feasibility study proved it is possible to automatically generate a union catalog with few mismatches or missed matches, and to create and update this catalog economically. (Author/JAB)

****
* Documents acquired by ERIC include many informal unpublished *
* materials not available from other sources. ERIC makes every effort *
* to obtain the best copy available. Nevertheless, items of marginal *
* reproducibility are often encountered and this affects the quality *
* of the microfiche and hardcopy reproductions ERIC makes available *
* via the ERIC Document Reproduction Service (EDRS). EDRS is not *
* responsible for the quality of the original document. Reproductions *
* supplied by EDRS are the best that can be made from the original. *
****

ED148340

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

A STATE-WIDE UNION CATALOG FEASIBILITY STUDY

Final Report on
Project III-FY 1976

for

The Illinois State Library

by

Martha E. Williams
Principal Investigator
and
Keith D. MacLaury

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Martha E. Williams

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

Information Retrieval Research Lab
Coordinated Science Lab
College of Engineering
University of Illinois
Urbana, IL

October 1977

IR005 441

Table of Contents

List of Tables and Figures		iii
Preface		iv
1.	Executive Summary	1
1.1	Test Files	1
1.2	Matching Duplicates	1
1.3	Bibliographic Data Elements Used for Matching	2
1.4	Development of IUCS Screens for Matching Records	2
1.5	Development of IUCS Procedures for Union Catalog Generation	3
1.6	Testing IUCS Screens and Procedures	4
1.7	Testing the Effectiveness of IUCS	4
1.8	Cost of Operating IUCS	5
1.9	Benefits and Future Requirements	5
2.	Defining Duplicate Records	7
3.	Preliminary Selection of Potential Bibliographic Data Elements for Inclusion in a Matching Key	9
3.1	High Frequency Fields (100-50%)	9
3.2	Medium Frequency Fields (49-25%)	10
3.3	Low Frequency Fields (24-0%)	11
3.4	Candidate Fields for Use in IUCS	11
4.	Rationale for Rejecting LC Card Number and ISBN Fields	14
5.	Literature Review of Bibliographic Keys	17
6.	Developing a First Pass Record Key	19
6.1	Overview	19
6.2	Title Keys	19
6.3	Notation Scheme and Conventions Employed for Keys	20
6.4	Testing Keys via Entropy Measurements	21
6.5	Title Key Development Based on Entropy Measures	24
6.6	Addition of Dates to the Title Key	27
6.7	Test Results for Candidate "Title-Date" (TD) Keys Using the LC MARC File	27
7.	Developing a Second Pass Screen	29
7.1	Author Matching	29
7.2	Full Title Matching Using Harrison Keys and Hamming Distance	30
7.3	Paging Matching	32
8.	IUCS Process for Generating A Union Catalog	34
9.	Full Scale Test of IUCS	40
9.1	Summary of Test Results -- Including Extrapolation to a Million Record Base File	40
9.2	Test Results for Merging OCLC:NW:UC Files	41

10.	Testing IUCS for Mismatches and Missed Matches	45
10.1	Mismatches	45
10.2	Analysis of Missed Matches	46
10.2.1	Variation in Titles	47
10.2.2	Variation in Date Data	51
10.2.3	Variation in Paging	52
10.2.4	Variation in Author Data	53
10.2.5	Summary	54
11.	Testing the Effectiveness of IUCS vs. LC/ISBN	55
12.	Computer Costs for IUCS Processing	56
12.1	Pre-Processing Costs	56
12.2	Pass 1 Processing Costs	56
12.3	Pass 2 Processing Costs	57
12.4	Post-Processing Costs	57
12.5	Estimated Cost of Adding Files to a MASTER file of One Million Records	58
13.	Benefit of Research Results and Future Needs	60
Appendix A	Technical Specifications	62
	Hardware	62
	Software	62
	Bibliographic Record Structures	62
	Record Key Format	63
Appendix B	Major Processing Programs	64
	OCLC, NW, UC (Format Conversion and Statistics)	64
	TTLRDR (Title Reader)	64
	COMEXT (Compare and Extract)	64
	PASS2	65
	RRREAD (Random Record Reader)	66
	TDMARK (Title-Date Marker), RRMARK (Random Record Marker), RRERAS (Random Record Eraser)	66
	CONCAT (Concatenate)	66
	KEYPRT (Key Printer)	67
	READER	67
Appendix C	Stopwords for Corporate/Conference Entries	68
Appendix D	Bibliography	71

Tables and Graphs

Table

1	High Frequency Fields	9
2	Medium Frequency Fields	10
3	Low Frequency Fields	11
4	Possible Candidate Fields for Use in IUCS	11
5	Field Frequencies for Sample Files	13
6	Records Containing LC and ISBN	15
7	Internal Matches and Mismatches Found Using LC and ISBN Codes	16
8	Characters Used for Encoding Keys	21
9	Candidate Title Keys	27
10	Paging Situations Found in Pairs of Records That are Potential Duplicates	33
11	Generation of MASTER (OCLC vs. OCLC) to Find Internal Duplicates -- Starting Conditions	42
12	Generation of MASTER (OCLC vs. OCLC) to Find Internal Duplicates -- Pass 1 Results	42
13	Generation of MASTER (OCLC vs. OCLC) to Find Internal Duplicates -- Pass 2 Results	42
14	Addition of NEW File (NW) to MASTER File (OCLC) -- Starting Conditions	43
15	Addition of NEW File (NW) to MASTER File (OCLC) -- Pass 1 Results	43
16	Addition of NEW File (NW) to MASTER File (OCLC) -- Pass 2 Results	43
17	Addition of NEW File (UC) to MASTER File (OCLC+NW) -- Starting Conditions	44
18	Addition of NEW File (UC) to MASTER File (OCLC+NW) -- Pass 1 Results	44
19	Addition of NEW File (UC) to MASTER File (OCLC+NW) -- Pass 2 Results	44
20	LC and ISBN Occurrences in OCLC and NW	46
21	Matching Effectiveness of IUCS-TD vs. LC vs. ISBN	55
22	Costs for Full Scale Tests on IUCS	57
23	Costs for Adding a 200,000 Record File to a MASTER File of 1,000,000 Records	59
24	OCLC Stopword List	68
25	Initial IUCS Stopword List	69
26	Most Frequent Initial Words After Stoplisting	69

Graph

1	Minimum and Maximum Entropy vs. % Unique Keys	23
2	Positional Entropies for a 44 Character Key	26

Preface

Staff

The Illinois State Library funded research project entitled "A State-wide Union Catalog Feasibility Study" was carried out within the Information Retrieval Research Lab (IRRL) of the Coordinated Science Lab (CSL) at the University of Illinois in Urbana-Champaign between May 1, 1976 and July 31, 1977. The research project was designed and directed by Professor Martha E. Williams, principal investigator, and the major research tasks were carried out by Keith MacLaury whose work on the development of a title-date key was used in partial fulfillment of the requirements for the M.S. degree in Computer Science at the University of Illinois. (Further details regarding the title-date key are provided in Mr. MacLaury's thesis entitled "An Investigation of Two Fixed-Length Keys for Variable Length Strings," December, 1977.) Several other members of the IRRL staff provided assistance in various tasks. They include: Scott Preece, Larry Lannom, Linda Scott, and Sue Lee.

Participating Libraries

In order to test merging and identification of duplicate records in multiple files we needed to obtain large sample files of records in computer-readable form. We were fortunate in obtaining the cooperation of the following four organizations: Northwestern University (through Velma Veneziano and James Aagard), the University of Chicago (through Charles Payne, Helen Schmierer, and Lenore Belsky), the Ohio College Library Center (through James Rush) and Northern Illinois University (through Robert Hamilton). We are most appreciative of the cooperation provided by these organizations in supplying test files and documentation about their data bases.

1. Executive Summary

The result of the "State-wide Union Catalog Feasibility Study" was the development of a prototype system called IUCS (IRRL Union Catalog System). IUCS was developed by testing various techniques for matching records against large test files of machine-readable records from major institutions that maintain their catalogs in machine-readable form. Overall the IUCS involves four major steps: (1) a pre-processing step which converts incoming files into IUCS form; (2) a first pass screen which matches records according to an efficient title-date matching key; (3) a second pass screen which provides more refined matching of the potential duplicates located in the first pass; and (4) a post-processing step which merges the new records into the existing MASTER file to create a union catalog.

1.1 Test Files

The machine-readable files of MARC-like records used to test the IUCS were provided by Northwestern University (NW), the University of Chicago (UC), Northern Illinois University (NIU), and the Ohio College Library Center (OCLC). The records from OCLC were records generated for one anonymous large university in the midwest. Records used for testing all aspects of the system included only those from NW, UC, and OCLC. NIU was unable to provide full bibliographic records because of technical difficulties. NIU experienced difficulty in converting their catalog card image records to MARC format. We wrote and tested a program that converted their catalog card images to a format suitable for use with the Library of Congress recognition program but apparently NIU was unable to get the LC program to run. Since we needed to be able to identify specific fields for the IUCS and since NIU was not able to provide the field identification we were unable to include the NIU file in the IUCS test. We did, however, use their records of LC card numbers in order to check LC number overlap between files.

In order to insure a reasonable degree of overlap between the records of the cooperating libraries we extracted a time-distinguished sample consisting of records for items entered into the files for the 1974-75 time period. The files used in the full scale test included 73,552 records from OCLC, 57,728 records from NW and 22,857 records from UC.

1.2 Matching Duplicates

The objective of the project was to develop an inexpensive and effective technique for matching records in multiple files in order to build a union catalog containing only one representation of a given book. Any system for matching records must have a definition of what a duplicate is. Our rules are slightly looser than the rules employed by strict catalogers, e.g., if title, date, author, and paging match, but the editions differ or if one is a hardcover and the other a paperback, we would still consider them duplicate records. Discussion of the IUCS definition of a duplicate is provided in Section 2.

1.3 Bibliographic Data Elements Used for Matching

Matching bibliographic records is dependent on the presence of data elements in the candidate files and the discrimination power of the data elements (fields) that are present. In order to determine which data elements should be used by IUCS we analyzed each of the test data bases and compared the field occurrences for each with the field occurrence data for the LC MARC data base. After analyzing fields with respect to their liklihood of occurrence and their ability to discriminate between records, we determined that the fields to be used for IUCS would be: subfields A and B from the title field, pagination from the collation field, and author(s) from the main and added entry fields. Details regarding the data element analysis are provided in Section 3.

One might expect LC card numbers or ISBNs to be useful identifiers for determining overlap (duplication) between files and that they would be possible candidates for initial screening. We tested these identifiers to determine their utility for union file generation. We found that they were not present in a sufficient number of records in the test files and they were not as accurate as IUCS. LCs occurred in 57% of the OCLC records for library X, in 87% of the NW records and in 75% of the UC records. ISBNs occurred in 41% of the OCLC records, 42% of the NW records and 28% of the UC records. since IUCS screening data are generated from the records and relate to 100% of the records, they provide a better basis for initial screening.

Even when LCs and ISBNs are present they produce a number of erroneous matches (mismatches) or false hits. We ran internal matches for each of the test files (each file against itself) to determine the number of records in a file having the same LC card number or ISBN. From the OCLC file of 73,552 records there were 42,001 records containing LCs and 30,548 containing ISBNs. Internal matching of the LC and ISBN files produced 271 and 263 internal matches. Of the 271 internal matches on LC, 19 were mismatches (they paired different records). Of the 263 internal matches on ISBNs, 41 were mismatches. From the NW file of 57,728 records, 50,508 contained LCs and 24,337 contaned ISBNs. Internal matching provided 843 internal matches on LC (holdings of separate NW libraries) of which 31 were mismatches and 486 internal matches on ISBN of which 49 were mismatches. From the UC file of 22,855 records, 17,250 were found to contain LCs and 6,386 contained ISBNs. Internal matching produced 17 internal matches on LC of which 7 were mismatches and 8 internal matches on ISBN of which all 8 were mismatches. ISBN mismatches appear to be due largely to the reuse of ISBNs by publishers. LC mismatches appear to be due largely to coding errors. Most of the mismatches in both cases were of unrelated works, e.g. the titles "Claude the Dog" and "Stretching Man's Mind" had the same LC card number. IUCS produces almost no mismatches. A manual examination of a large number of pairs of records matched by IUCS produced only two mismatches and they were of records that were very closely related bibliographically (a teacher and student edition of the same text and two translations of the same work). Discussion regardng the analyses of LCs and ISBNs as possible candidates for matching is provided in Section 4.

1.4 Development of IUCS Screens for Matching Records

IUCS was planned to be a multiscreen system for matching records in which the first screen would be a broad screen that would pull out the candidate duplicate records. A set of finer screens would be used for verifying pairs of records as duplicates, rejecting pairs as non-duplicates or indicating a need for manual checking to verify or reject pairs.

Development of the broad screen or first pass in the IUCS involved the design of a fixed-length code based on information from the titles and dates of the records in the files to be matched. This code, called a matching key, should be efficient and effective. That is, it should provide a fast, inexpensive match and it should not fail to match records that are the same. Various key elements from the title portion of records were selected and tested, using entropy measures to determine the discrimination power of the key, and candidate title keys were determined. Although the title keys would work well in most cases, it was necessary to add "date" to the key in order to discriminate among multiple identical titles such as "Proceedings" or "Collected Works." The result of tests on title-date keys produced the title-date key which is based on selected alpha characters from the beginning and end of titles plus date information.

Preparatory to development of the title-date (TD) key we analyzed relevant papers discussing search keys and matching keys developed by other researchers. Results of the literature review are given in Section 5. A discussion of the development of the TD key is provided in Section 6. Additional details regarding the TD development are presented in Mr. MacLaury's thesis as this portion of the research program provided the subject matter for his thesis.

The TD key is used for the first pass matching of files. Results of this matching step are then subjected to finer screening in a second pass procedure. the second pass includes three procdures: (1) matching of authors; (2) matching of titles using Harrison keys (which are bit string representations of the titles) and Hamming distances; and (3) matching of paging. The author match allows any combination of main and added entry. Harrison keys, which are bit strings representing the character strings of the titles, are generated for each title. The Harrison keys for two records, which are candidate duplicates, are then compared. The comparison does not involve a character by character comparison but indicates the extent of differences between titles. The comparison involves calculating the Hamming distance between title bit strings. Hamming distance is a representation of the differences in the two bit strings derived from the titles. It is the sum of the number of instances of non-match in the bit strings. The more differences there are between two titles, the greater the Hamming distance will be. By allowing Hamming distances up to eight for the IUCS second pass screen, we were able to overcome small variations in the recording of titles and a certain number of typographical errors.

The third procedure involved in pass 2 is the match on paging. There again we did not want to require exact matches but matches within a range. This is done because of the variant ways in which pages are counted and recorded. Details regarding the second pass screen are given in Section 7.

1.5 Development of IUCS Procedures for Union Catalog Generation

After completing the analyses and tests that led to the development of the first and second pass screens, we developed an entire set of procedures that would be needed for generating a union catalog. The entire set of nine procedures fall into four major groups: (1) pre-processing, (2) pass 1, (3) pass 2, and (4) post-processing procedures. Pre-processing involves the translating of incoming files to the IUCS format. Pass 1 involves creation and comparison of title-date (TD) keys. Pass 2 involves: accessing the records associated the the TD keys that matched; determining and marking duplicates; printing out records that require manual examination; entering the results of the manual examination; and sorting the accepted TD keys (those that represent new records). The post-processing steps involve adding the new records that do not duplicate items in the MASTER catalog file to the MASTER file, and adding the TD keys for the new records into the file of TD keys for the MASTER file. Every time a new file (representing a new library) is added to the union catalog (MASTER file) the processing procedures are repeated. Section 8 provides a description of IUCS processing.

1.6 Testing IUCS Screens and Procedures

After developing the full set of procedures involved in generating a union catalog, we tested the procedures by generating a union catalog of the sample files from OCLC, NW, and UC.

The largest sample file (OCLC with 73,552 records) was selected as the MASTER file to which NW and UC were added. OCLC was first matched against itself to locate internal duplicates. This produced 702 internal duplicates leaving 72,850 records to function as the MASTER file. The NW file (57,737 records) was then checked against itself producing 728 internal duplicates and 57,009 records to be matched against the OCLC (MASTER) file. this match of NW vs. MASTER indicated that 14,218 of the NW records already were in the MASTER file, thus the remaining 42,791 NW records were added to the MASTER file. At this point MASTER contained 115,641 records. The next file to be added was UC. UC had 22,857 records of which 42 were internal duplicates leaving 22,815 to be matched against the MASTER file. Of the 22,815 we found 5,130 to be duplicated in the MASTER file, thus 17,685 UC records remained to be added to the MASTER file. At the end of this merge, MASTER (OCLC+NW+UC) contained 133,326 records.

In addition to providing the real data regarding the cross matching to generate a union catalog of records from three files, we extrapolated all figures based on a representation of OCLC file as having a million records. A summary of the findings of this test and the extrapolation data are given in Section 9.

1.7 Testing the Effectiveness of IUCS

The adequacy of any scheme for creating a union catalog can be measured in terms of the two cardinal errors that may occur. Records that do not relate to the same item may be erroneously identified as duplicates -- these are mismatches -- and records that truly relate to the same item may not be identified as duplicates -- these are missed matches. Of the two types of errors, mismatches are more serious because they result in information being permanently lost. If a new record is

mismatched (said to match even though it does not) with a record in the MASTER catalog it is then discarded because it is assumed not to be a new record. In order to determine the mismatch rate of IUCS we selected 3,686 record pairs that had been matched by IUCS. Manual examination of the 3,686 pairs indicated that two records were mismatched and they were very closely related bibliographically. One was a result of a cataloging error and hence is not really an error as far as IUCS is concerned. The other is a duplicate according to the IUCS definition. A minor change in IUCS would accomodate this type of mismatch getting through. Thus the mismatch rate of IUCS, based on examination of 3,686 records is .054% and with a minor change in the system (checking the authorship statement or the 700 fields for indications of translation) it would be reduced to almost zero. Simple screens such as LC matches, ISBN matches or use of search keys produce considerable numbers of mismatches.

Errors due to missed matches result in an unnecessary expansion of the union catalog. These are less serious in that information is not lost, it is merely unnecessarily recorded twice.

In order to determine the percentage of matches missed by IUCS, we used all possible means for matching records in the OCLC and NW test files and developed a set of duplicates that, as far as we could determine contained 100% of the true duplicates in the file. Based on matching within this set, the effectiveness ratings (in percentages) of LC, ISBN, IUCS-TD, and combinations were: LC, 74.90; ISBN, 56.58; IUCS-TD, 97.10; LC/ISBN, 80.74; IUCS-TD/ISBN, 98.17; and IUCS-TD/LC/ISBN, 99.62. Missed matches were missed for a variety of reasons. In a similar test of a restricted file, the distribution of missed matches fell into four categories. Variations in titles (addition or substitution of words, use of German abbreviations, typographic errors, etc.) accounted for 81% of the errors. Variation in year of publication data accounted for 12.3% of the errors. Variation in paging accounted for 5.8% of the errors and variation in author data accounted for <1%. Sections 10 and 11 provide discussion of tests for IUCS effectiveness.

1.8 Cost of Operating IUCS

An important foctor in evaluating any system is the cost of running the system. Our programs were written in the SAIL language and were run on the CSL's DECsystem-10. Computer costs for the full scale preparation and merging of records from OCLC, NW, and UC were recorded. Costs are related to the four major processing steps: pre-processing, pass 1, pass 2, and post-processing. Using nighttime rates (which are approximately 10% of day rates), the cost of pre-processing was $20.68, the cost for pass 1 processing was 8.62, the cost of pass 2 processing was $11.57, and the cost for post-processing was $1.45, for a total cost of $42.20.

Because our runs involved 154,000 records we extrapolated our actual costs to simulate a comparison of 200,000 records in a "new" file with a million record MASTER file. Assuming a 25% overlap, the nighttime cost would be $70.90 (or $592.55 during the day). If the overlap were 50% the cost for nighttime processing would be $96.88 (or $769.49 during the day). On top of these costs one would have to add the personnel cost for manual checking of 1.1% of the duplicates located. Approximately 1.1% of duplicates are pairs that IUCS prints out for manual verification. Details regarding cost of the IUCS runs and extrapolation to the million record file are given in Section 12.

1.9 Benefits and Future Requirements

The feasibility study has proved that it is possible to automatically generate a union catalog from non-standardized machine-readable files. It is possible to do so effectively (very few mismatches and missed matches) and efficiently (for a low cost and with little human intervention). IUCS can create the initial catalog inexpensively and update it -- adding new library holdings -- inexpensively. Such a scheme can be used on a state-wide (multi-institution), an institution-wide (multiple libraries within an institution), and on a regional or inter-state basis. It provides a capability for enhancement of records from contributing libraries with subsequent generation of machine-readable files for those libraries. It provides a basis for record and subject searching (subjects via the Harrison keys -- bit strings representing full titles). If IUCS were implemented on a full scale basis and the files were maintained on-line, it would provide an efficient basis for on-line searching for interlibrary loan. IUCS provides not only a capability for generating union catalogs -- as would any matching scheme -- but it does so with a high degree of accuracy. Accuracy is provided because IUCS algorithms can: handle records that are formatted differently; disregard differences based on diacritical marks; overcomes variations in cataloging practices; overcomes many differences in records that result from spelling errors or English vs. U.S. spellings; overcomes many typographical errors; and accomodates different representations in problematic corporate entries, such as U.S. vs. US vs. U. S. vs. United States.

The study was a research study to determine the feasibility of automatically generating a union catalog. The feasibility has been demonstrated. A few modifications could be made to increase the effectiveness of the screens but the limitations of time and money prevented this. If the results are to be exploited for the State of Illinois, it would be necessary to make minor modifications in the systems and to program the system in a higher level language for a popular computer system. Further work would be required in reprogramming and documentation in order to generate a transportable "turn-key" system.

2. Defining Duplicate Records

To build a union catalog of bibliographic records, criteria are needed for determining whether a new record matches a record already in the file. In the past this was probably an implicit aspect of union catalog maintenance. The clerks or librarians filing new cards made this judgement on a case by case basis. Their decisions were probably based on the rules used in the local catalog department for determining whether a new item was an added copy. These rules generally separate (i.e., do not call duplicates) any two items or records that differ at all in the descriptive cataloging. Thus separate editions (perhaps including paperback editions), reprints, and perhaps even items that only differ slightly in paging may be separated. One reason these rules have been strict is that it eases the decision process.

These strict bibliographic rules, while suitable for union card catalogs have been "loosened" for this project for the following reasons. First, it has been estimated that 90-95% of all users of an interlibrary loan network do not care which edition of a book they get or whether the book is the original edition, a reprint, or a paperback. Furthermore, while space requirements may not be an overriding consideration in a union card catalog, file size for a machine-readable file is a much more important consideration. Even if not maintained on-line, processing and printing costs are affected by file size and on-line storage costs, despite all technological advances, are still high. The more duplicates found, the fewer separate records in the combined file. Thus, strict bibliographic standards of what constitutes a duplicate have been modified as follows for the IUCS algorithms:

1. Date matching: Two records may match if any of the dates in the collation match. Thus, two editions having "1970" and "1976 [c 1970]" would be called duplicates if they met the other criteria.
2. Variant paging: Two records may be matched if the difference between their primary pagings falls within the guidelines of the IUCS paging test (see Section no. 7.3).
3. Location and name of publisher, size, edition: Differences in these elements are ignored. Once the other criteria have been met, these elements only serve to differentiate variants like American versus English edition or hardbound versus paperback edition.

In addition to these departures from strict bibliographic matching, variances from exact match that are generally allowed in cataloging department determinations of duplicates have been implemented. These include:

1. Variation in analytic cataloging: Two records may match if the main entry from one matches an added entry from the other. Two records matching only on added entries are examined manually for possible duplication.
2. Variation in title recording: The IUCS algorithms allow records with slight variations in the recording of the titles to match. Generally this encompasses typographic errors and

will not match two records only one of which has a long subtitle included (see Section no. 7.2).

These criteria have been developed through testing of sample files and examination of record pairs matched.

3. Preliminary Selection of Potential Bibliographic Data Elements for Inclusion in a Matching Key

A necessary condition for a data element to be used in matching is that it be present in a sufficient number of records. Otherwise its discrimination value can be presumed to be negligible. A recent study by this lab of one year's worth of MARC II records (1974-75) [Williams and Shefner] provided the figures needed to form a preliminary list of usable data elements. To the extent that the records from the participating libraries would be either derived from MARC records or based on the same cataloging rules, these figures were presumed to be indicative of what to expect on the sample tapes.

The following tables show the most common MARC fields and the percentage of the 124,355 records studied in which they occurred. Certain ranges of tags have been grouped where the fields serve the same function relative to the record and differ only in their internal content.

3.1 High Frequency Fields (100-50%)

The following fields were found to have a high frequency of occurrence in LC MARC records:

Tag(s)	Name	%	Reject/Accept	
600-651	Subject Headings	100.00+*	R	
245	Title	100.00		A
260	Imprint	99.99		A
050	LC Call No.	99.93	R	
300	Collation	99.82		A
082	Dewey Classif. No.	96.66	R	
100-111	Main Entry	93.19		A
020	ISBN	55.08	R	

Table no. 1 High Frequency Fields

The LC call number (tag 050), the Dewey classification number (tag 082), and the Topical Subject Heading (tag 650) on LC MARC records can all be considered as "suggested" subject classifiers. Subject classifiers are relatively non-specific as compared to fields such as author or title, in addition, they are subject to frequent and inconsistent alteration by user libraries. It is even less likely that locally cataloged items would match LC cataloging in these fields. Thus it can be presumed that these fields would, despite their frequency of occurrence, be poor indicators for identifying duplicate items.

ISBN and LC card numbers were rejected for inclusion in the search key because at the time it was presumed that they would be used separately to find duplicates in a preliminary matching step. Unfortunately they were found to be considerably less reliable than

*Summing of individual fields is inaccurate because of multiple occurrences.

anticipated (see Section no. 4). LC card number statistics were not included in this study since they are presumed to appear in all records.

The remaining fields -- title, imprint, collation, and main entry -- are fairly specific, and, in general, consistently applied by a variety of libraries. It was presumed that they would supply most of the elements of our search key.

3.2 Medium Frequency Fields (49-25%)

The following fields were classified as occurring with medium frequency in the LC MARC sample:

Tag(s)	Name	%	Reject/Accept
504	Bibliography note	47.75	A
700-740	Added entries	43.17*	A
500	General notes	41.72	R
043	Geographic Area Code	40.64	R
400-490	Serials	32.28	A
350	Bibliographic price	25.74	R
015	National Bibliographic No.	25.57	R

Table no. 2 Medium Frequency Fields

Several of these fields may be rejected for use in a search key because they are unlikely to appear in the local cataloging records of the various libraries. These are the Geographic area code (813), Bibliographic price (350), and the National Bibliographic number (015). Subsequent analysis of the test files may prove this analysis false for the first two of these, but the third is strictly a result of LC's shared cataloging arrangement with the National Bibliographies of other countries.

The general note field (500) can be presumed to vary widely in its content just as the subject field mentioned above.

Medium frequency fields that are accepted for use in a search key can be used in different ways. The added entries (700-740), since they constitute variant entry points, can be considered the basis for constructing alternate matching elements for the same record. For instance, a collection of articles with an editor may be entered under title by library "A" and under editor by library "B". Unless the matching key ignores main entry, these records may fail to match unless the added entry for the editor which was presumably included in the record from library A is considered. This was suspected to be a common variation in cataloging practice and one that could not be caught by single key/record methods.

The two other fields from this collection that might be considered for use in a search key are the Bibliography note (504) and the Series statement (400-490). These fields might be used to reject potential matches if they were present in both records and differed significantly in their content. This might be done in the context of a second pass "filter" that constructed new keys from the records of suspected

*Summing inaccurate due to multiple occurrences.

duplicates to eliminate false duplicates.

3.3 Low Frequency Fields (24-0%)

The following fields were found to have low frequencies of occurrence in our sample LC MARC file:

Tag(s)	Name	%	Reject/Accept
250	Edition statement	14.324	A
800-840	Series added entry	8.710	A
041	Languages	7.455	R
520	Abstract or annotation	5.645	R
240-241	Supplied title	5.140	A
505	Contents note	3.878	A
060	NLM Call no.	3.072	R

Table no. 3 Low Frequency Fields

All of the accepted fields here (except 240-241) could possibly be used in a second pass filter. The supplied title fields (240-241) could be treated as variant titles and be used to generate extra keys. The Abstract or annotation field (520) will be every bit as variable in its content as the General note field. And the NLM Call number is another subject classifier (and one not likely to be used by general libraries because it only would apply to medically related documents). The Languages field (041) probably does not appear in local cataloging records.

3.4 Candidate fields for Use in IUCS

Based on this analysis, a set of "reasonable candidate" fields remained for use in a first pass rough screen and a second pass fine screen:

First pass		Second pass	
245	Title	504	Bibliography note
260	Imprint	400-490	Series
300	Collation	250	Edition statement
100-111	Main entry	800-840	Series added entry
700-740	Added entries	505	Contents note
240-241	Supplied titles		

Table no. 4 Possible Candidate Fields for Use in IUCS

Subsumed within some of these fields are subfields that need to be considered separately in order to complete an analysis of potential elements for a matching key. These include short title, subtitle, place of publication, publisher, date of publication, date of copyright, paging, and size. The Williams and Shefner study did not generate statistics at the subfield level so no further consideration of these elements on a frequency basis was done. In all likelihood they are

present most of the time.

Of these possible candidate fields, those actually used in IUCS were subfields a and b from the title field, date from the imprint field, pagination from the collation field, and authors from the main and added entry fields.

Table no. 5 gives the actual field frequencies found in our three sample files along with their average and the frequency of that field in the LC MARC 1974-75 file.

Field	Tag	NW	OCLC	Chi	Avg.	MARC
Control No.	001	100.00	100.00	100.00	100.00	100.00
Fixed Length Data	008	100.00	100.00	100.00	100.00	100.00
*LC Card No.	010	87.40	57.00	75.48	73.29	
National Bibl. No.	015		7.63	2.77	3.47	25.57
*ISBN	020	46.74	41.53	31.27	39.85	55.08
Overseas Acquisition	025		0.02	0.02	0.01	0.25
Cataloging Source	040		7.84	99.96	35.93	0.08
Languages	041	4.80	5.82	3.78	4.80	7.46
Geo. Area Code	043		13.59	7.91	7.17	40.64
LC Call No.	050	79.87	55.57	6.34	47.26	99.93
Copies, issues	051		0.05	0.04	0.03	0.33
NLM Call No.	060		2.95	0.04	1.00	3.07
NAL Call No.	070		0.22	0.00	0.07	0.02
Dewey Classification	082	61.08	41.86	1.36	34.77	96.66
SuDocs No.	086		0.12	0.00	0.04	0.89
*	100	84.90	83.97	81.15	83.34	79.45
*	110	5.44	5.89	9.09	6.81	11.57
*Author	111	1.83	2.25	2.46	2.18	2.18
*	130	0.28	0.30	0.47	0.35	0.38
*	240	0.24	1.51	3.32	1.69	5.13
*Title	241		0.01		0.00	0.01
*	245	100.00	100.00	100.00	100.00	100.00
*Edition	250	13.65	15.41	15.73	14.93	14.32
*Imprint	260	99.99	99.99	100.00	99.99	100.00
*Collation	300	99.77	99.78	100.00	99.85	99.82
*Bibl. Price	350		5.56	39.17	14.91	25.74
*	400	0.01	0.07	0.00	0.03	0.10
*	410	0.75	1.42	0.50	0.89	1.95
*Series	411		0.02		0.01	0.02
*	440	10.25	7.42	1.93	6.53	8.08
*	490	26.56	23.76	6.16	18.83	22.14
General Notes	500	43.94	37.95	36.39	39.43	41.72
"Bound with"	501	0.21	0.01	0.10	0.11	0.06
Dissertation	502	0.63	4.37	2.47	2.49	0.39
*bibliography	504	60.28	52.99	54.92	56.06	47.75
*Contents	505	3.38	3.51	3.73	3.54	3.88
Abstract	520	1.36	0.26		0.54	5.65
	600	16.02	13.10	15.13	14.75	11.55
	610	4.58	3.21	4.47	4.09	5.16
Subject Headings	611	0.09	0.06	0.10	0.08	0.11
	630	0.51	0.66	0.87	0.68	0.84
	650	66.63	68.39	75.30	70.11	73.14
	651	21.42	14.39	4.46	13.42	16.31
*	700	27.63	25.62	25.56	26.27	27.50
*	710	8.48	8.36	8.81	8.55	10.11
*Added Entries	711	0.23	0.32	0.33	0.29	0.28
*	730	0.66	0.61	0.94	0.74	1.06
*	740	3.37	4.29	5.11	4.26	4.24
*	800	0.08	0.04		0.04	0.07
*Series Tracing	810	6.30	3.69	1.17	3.72	6.67
*	811	0.03	0.02	0.02	0.02	0.02
*	840	1.62	0.96	0.31	0.96	1.95

Table no. 5 Field Frequencies for Sample Files

* Fields kept for IUCS records

4. Rationale for Rejecting LC Card Number and ISBN Fields

Originally we had planned to find duplicate records using LC card numbers and/or ISBN's as the initial step of the IUCS (IRRL Union Catalog System). It was expected that this procedure would save processing time since it would reduce the amount of work to be done by IUCS by finding many of the easy matches quickly. This approach, however, has not proved feasible in our experiments for three reasons: it does not save that much computer processing time; it is not as accurate as the IUCS; and the numbers were found to be present in 57-87% (LC) and 28-42% (ISBN) of the records as shown in table no. 6.

Although our cost estimates indicate that there would be some savings when there were large numbers of duplicates, the savings resulting from matching LC card no. and ISBN were mostly offset by the cost of generating the LC and ISBN keys.

The accuracy problem was most apparent in tests for internal matches (Matching records within a single file), and less evident in tests involving cross-file matches.

In six different tests for internal matches using LC and ISBN keys, the percentage of pairings that were mismatches (two records having the same LC or ISBN but which represented different works) ranged widely (see table no. 7). These matching failures were located by running the PASS2 program (see page no. 65) against records with matching control numbers and then examining the record pairs that failed PASS2. This examination revealed two major categories of matching failures:

1. Unrelated works. 73% of the mismatched pairings were bibliographically unrelated works. For example, two OCLC records had the same LC card number (74-000078). One was David Farrer's The Warburgs: The Story of a Family and the other was Samuel Tisdale's Soil fertility and Fertilizers Another example of two records with the same LC card number for different monographs is the following:

```
010  74003403
020  0884052826
100  HARMON, MARGARET.
245  STRETCHING MAN'S MIND :  $B A HISTORY OF DATA PROCESSING / $C BY
     MARGARET HARMON.
260  NEW YORK :  $B MASON/CHARTER, $C 1975.
300  XV, 239 P.  :  $B ILL.  ;  $C 22 CM.

010  74003403
020  0816431167
100  GACKENBACH, DICK.
245  CLAUDE THE DOG;  $B A CHRISTMAS STORY.  $C WORDS & PICTURES BY DICK
     GACKENBACH.
260  NEW YORK, $B SEABURY PRESS $C [1974]
300  [32] P.  &B COL. ILLUS. $C 18 X 24 CM.
```

The major reason for LC card no. mismatches is coding errors. In the case of ISBNs, the check digit reduces the possibility of coding error, and the cause of the mismatches seems to lie with the publishers. In nearly all cases of ISBN matches for bibliographically unrelated

works, both works in the pairing came from the same publisher, implying reuse of ISBNs by the publisher. Many different publishers, including several large ones, showed up in our examples.

2. Related Works. 27% of the mismatched pairs consisted of bibliographically related items such as a set matched with a single volume from the set, a main volume matched with its supplement or index volume, or two different issues of an annual being matched. For example, American Library Laws and its Supplement having the same ISBN.

Since there is no provision for dash-on information in the current MARC format, the use of the same LC card no. on two different MARC records is legitimate when one would normally be a dash-on record. However, other LC matches of related works, and matching ISBNs, are errors, presumably originating when the publisher prints the same control numbers in different volumes of a set, different issues of an annual, different editions of a work, etc. This error is propagated by catalogers who copy the numbers and by data base systems that do not check for duplication.

All these cases of mismatch, regardless of cause, are unacceptable for use in the automatic generation of a union catalog since records that are truly different would be discarded. The system would assume that they were duplicates of records already in the catalog. The IUCS, by matching several different record elements, avoids mismatches almost entirely. In a manual examination of 2,838 pairs of records matched by IUCS, only two (0.0704%) were found to be mismatches; and they were very closely related items.

File	Records	Records Containing LC	(%)	Records Containing ISBN	(%)
OCLC	73,552	42,001	(57.1)	30,548	(41.5)
NW	57,728	50,508	(87.5)	24,337	(42.2)
UC*	22,855	17,250	(75.5)	6,386	(27.9)
NIU	79,205	79,205	----	-----	----

* The records from UC represent a sample of Chicago's cataloging for 1974-75.

Table no. 6 Records Containing LC and ISBN

An examination of the pairs resulting from internal matches of LCs and ISBNs in the three test files produced the following numbers of internal matches and mismatches within the internal matches.

File	Code	Internal Matches	(% of Codes matching)	Mis-matches Related	Unrelated	Total
OCLC	LC	271	(0.6)	2	17	19
	ISBN	263	(0.9)	2	39	41
NW	LC	843	(1.7)	22	9	31
	ISBN	486	(2.0)	14	35	49
UC	LC	17	(0.1)	2	5	7
	ISBN	8	(0.1)	--	8	8

Table no. 7 Internal Matches and Mismatches Found Using LC and ISBN Codes

The problems with control number matching were much less evident with matches between files. The reason for this is unclear. The records in the OCLC and NW files (after removal of internal duplicates) that had LC card no. or ISBN fields were matched on these numbers and the resulting matches were again tested using the PASS2 program. (This test was also useful for estimating the number of duplicates missed by IUCS -- see Section no. 10.2.) Of the 11,823 matched record pairs matched by these control numbers, only 79 (.69%) were not duplicates. Of the 79 mismatches 21 matched on LC card no. and 58 matched on ISBN. Of the 79 pairs, 15 were bibliographically unrelated (11 pairs were volume versus set or series, and 4 were variant editions) and 64 were unrelated works. Of these unrelated non-duplicates, 47 were from the same publisher and 17 were from different publishers.

5. Literature Review of Bibliographic Keys

The literature on data dependent keys to represent machine-readable records in bibliographic files dates from 1968. Most of the keys covered in the literature were used as (or proposed as) search keys to locate and retrieve machine-readable records corresponding to a given book or book citation. When machine-readable bibliographic records are held in random access files, this searching capability is needed by acquisitions librarians verifying book orders, catalogers seeking cataloging copy for a book in hand, and by circulation and interlibrary loan clerks (or library patrons) seeking a book to borrow.

The earliest work in the area of search keys was based on the application of data compression techniques to the author and title strings [Coe; Dolby; Lipetz, et al; Newman and Buchinski; Nugent; Payne; Ruecking] with other record elements occassionally included. However, Kilgour, et al. soon dominated the field with the truncated keys developed for the Ohio College Library Center system. [Kilgour, et al; Landgraf, et al; Long, et al] Kilgour had two convincing arguments for the use of straightforward simple keys: 1) the key would have to be formulated and input many times by many people and thus should be easy to generate; and 2) the uniqueness was not important in an online system since several abbreviated records could be displayed on one or two screenfulls allowing the searcher to choose from the displayed "mini-catalog." The basic idea of truncating leading words from author and title has been widely adopted [Chauveinc; Inovar Corp.; Plaister, et al; Veneziano]. With the general acceptance of this particular key formation methodology, research turned from the development of keys to the analysis of key performance [Bookstein; Guthrie and Slifko; Kjell; Lowe].

However, the keys used by OCLC have already been lengthened once due to the increasing size of the OCLC base and the consequent unwieldiness of many mini-catalogs. An even larger title key has been proposed [Legard and Bourne]. Recently, more sophisticated search systems have reduced or eliminated the need for distinct search keys. One method is to combine a search key search with a second level search of the resulting set. This method was proposed by Bookstein [Bookstein] and has been implemented in a Bro-Dart search system where the user inputs a truncated title key and any key word from the desired title. The key is used in searching the entire file and the resulting set is searched using title word. Search keys have been completely eliminated in the BALLOTS system by fully inverting the words from the author and title fields. Thus the use of very short distinctive record keys for record retrieval will probably diminish in future systems.

However, searching and retrieval are not the only application for bibliographic record keys. Project LOC [Jolliffe; LOC Project] investigated bibliographic keys as a device for creating a union catalog of pre-1801 books in the collections of the British Museum, Oxford, and Cambridge. More recently, Ayres has proposed [Ayres] and he and others are developing and testing [Beale and Lynch; Ayres] what he calls a Universal Standard Book Number (USBN). His proposal is to develop an algorithm that would generate a unique ten digit number for the bibliographic data for a bibliographic item. He feels that this method would provide a more useful control number than the ISBN which is

assigned by publishers and not always reliable. The basic ideas contained in these two projects are closely related to the goals and methods of this project.

6. Developing a First Pass Record Key

6.1 Overview

The first process in finding duplicate records in combined files consists of extracting, sorting, and comparing keys from the records. These keys should be fixed length for economy and ease of sorting. The comparison step results in groupings of matching keys which represent potential duplicate records. These records can then be tested further in a second pass to determine which ones are really duplicates.

In developing a fixed length key for the first pass, the two major goals were to not miss potential duplicates and to partition the file as fully as possible (i.e. obtain the minimum number of minimum size groups) to minimize the work in the second pass. These two goals conflict since the more record elements used in the key, the better the partition and the greater the liklihood of missing duplicates due to variant cataloging practices and typographic errors.

6.2 Title Keys

The title field (subfields a and b) was chosen as a primary source for the key because it:

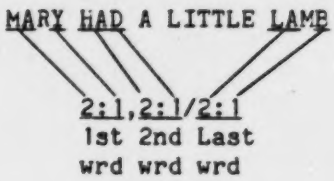
1. Is generally a good discriminator between records.
2. Is one of the fields least subject to variation in catalcging practice.
3. Appears in almost all the records (see Table no. 5 and [Legard and Bourne]).

Several methods of forming fixed length keys from variable length strings (e.g. titles) have been suggested and/or tried (see Section no. 5). They generally fall into two categories: concatenating selected characters and general mapping functions operating on the entire string (or large portions of it). The selection process for characters to be concatenated has been based on character position (e.g. "the first three characters of the first three words" or "every second character") or character types (e.g. the first nine consonants). Mapping functions can map strings to other alphabetic strings (Soundalike codes [Dolby]), numeric strings (Variable generator codes [Lynch]), or bit strings.

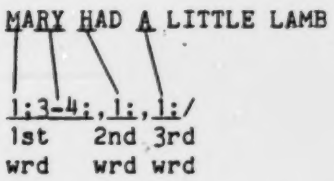
In order to maximize the error tolerance of the key and to deal effectively with the variable length aspect of title strings, character position keys were chosen for further study. The goal was still to partition the records as effectively as possible while involving as few typographic errors as possible. With character codes, this latter goal is approached by minimizing the number of character positions to be concatenated (while mapping keys would tend to not match exactly if the titles they were derived from differed at all). The first goal was the subject of extensive research. Based on other work, [Coe; Fokker and Lynch] it was decided to search for an optimal key derived from both the front and back of words and titles. A single figure measure of key effectiveness called entropy was used to develop the key.

6.3 Notation Scheme and Conventions Employed for Keys

In order to describe the title key and its development conveniently, a notational scheme was developed to describe the keys. The notation developed is an elaboration of the simple (and now well-known) search key notation used by OCLC. In our notation, colons separate the front and back of words, commas separate words, and the slash separates the front and back of titles. The numbers generally indicate the number of characters taken from the position implied. In this notation, the OCLC title key, commonly denoted as the 3,2,2,1 key would become 3:0,2:0,2:0,1:0/0 or 3:,2:,2:,1:/. As another example, 2:1,2:1/2:1 denotes a key formed by concatenating the first two letters and the last letter of the first two words and the last word of the title. For example, Mary Had a Little Lamb becomes MAYHADLAB.



In considering various types of keys there were some that were not made up of consecutive letters from the front or back of words. In these cases the notation used to indicate the location of the discontinuity is a semi-colon. The numbers after a semi-colon indicate character positions rather than the number of characters taken. A dash indicates contiguous locations. For example, 1;3-4:,1:,1:/ designates a key consisting of the first, third, and fourth letters of the first word and the first letters of the second and third words or:



As other examples, 1;3;5:/ would designate the first, third, and fifth letters of the first word; 2;4-5:/ indicates the first two and fourth and fifth letters. When it is necessary to skip positions from the end of a word, "L" (for Last) is used. Thus 1;3-4:3L-2L/ is a key consisting of the first, third, fourth, third from last and next to last characters of the first word.

:	Separates front and back of words
,	Separates words
/	Separates front and back of titles
#	Indicates the number of characters from indicated position or character position after a discontinuity
;	indicates a discontinuity
-	indicates contiguous letters after a discontinuity
L	indicates character position from the end of a word

Table no. 8 Characters used for Encoding Keys

Some other procedural conventions used in this development work (and in the IUCS) should be noted:

1. All letters were capitalized.
2. Hyphenated words were treated as one word and most other punctuation was suppressed. (The allowable character set contained 43 characters: A-Z,0-9,!,#,$,%,&,?, and blank.)
3. Only leading English language articles (A,An,The) were skipped before forming the key.
4. When characters/words were taken from both the front and back of words/titles, short words and titles were "overlapped" for key formation.

As examples of this fourth point (using a 2:1,2:1/2:1 key): The title The Americans becomes AMS AMS; and the title Alliance in Decline: A study in Anglo-Japanese Relations, 1908-32 becomes ALEINN192.

6.4 Testing Keys via Entropy Measures

Entropy measures based on information theoretic principles [Shannon] provide the best available single figure measure for determining the discrimination power of a given key (i.e. how effectively it partitions the associated records). It can also be scaled quite readily to a [0,1] scale. The use of entropy for measuring key effectiveness was first used by Fokker and Lynch in their work with variety generator search keys for author names [Fokker and Lynch].

For example, suppose the first two letters of the first non-article word was to be used as a key for a file of 100 titles. Let us first examine the two extreme cases:

1. All titles begin with the same letter pair (low entropy, poor discriminator)
2. Each title begins with a different letter pair (high entropy, good discriminator)

An entropy measure for keys is:

$$-\sum_{i} p_i \log_{fs} p_i$$

where p_i is the probability of the ith character pair occurring and fs is the file size.

Presuming the 43 element character set mentioned above, the entropies for the two extreme cases are:

1. $$-\sum_{i=1}^{43^2} p_i \log_{100} p_i = -1 \log_{100} 1 = 0$$

2. $$-\sum_{i=1}^{43^2} .01 \log_{100} .01 = -100(.01 \log_{100} .01) = 1$$

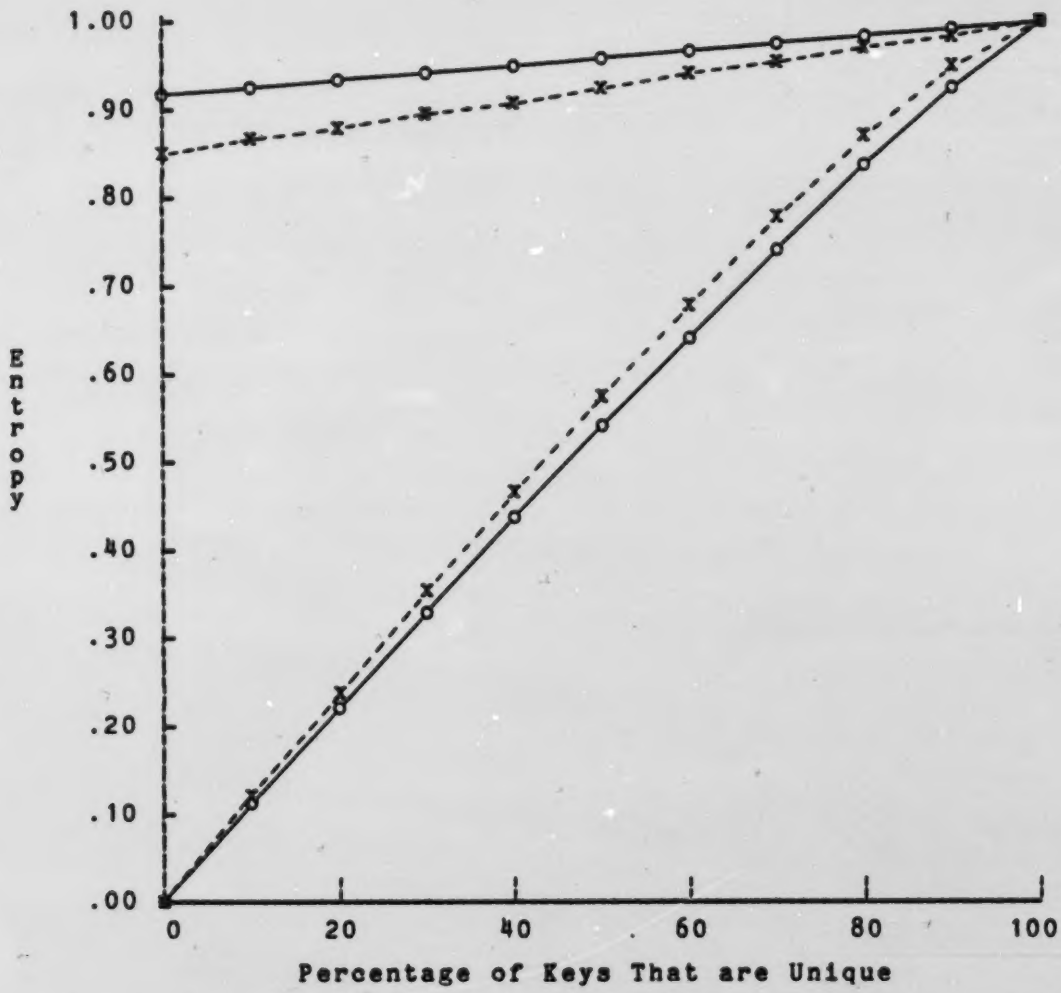
In this example, both summations would obviously have many zero elements, but it is clear that this method provides a scaled single figure measure for the dispersion of keys.

One consequence of this normalization is that keys whose complete set of values is smaller than the size of the file under consideration have an (easily calculated) upper limit on their entropy. For example, suppose the key used to identify a hundred titles was the first letter instead of the first two letters. Then the maximum obtainable entropy would be approximated by:

$$-\sum_{i=1}^{43} 1/43 \log_{100} 1/43 = \log_{100} 1/43 = .817$$

This upper bound varies with file size: for a file of 5,000 records it is 0.442, for 20,000 records, 0.380, and for a file of 100,000 records, 0.327.

Entropy takes into account not only how many keys are unique (an underlying assumption is that the records are unique) but also the "dispersion" of the remaining matching groups. The following graph shows the relation between entropy and the percentage of keys that are unique. The lower lines are the minimum entropy figures for files of 100 and 5,000 records. They represent the entropy when the non-unique portion of the key file consists of a single key (e.g. ABCDEEEE). The upper lines are the maximum entropies. They presume that the non-unique keys are grouped in pairs (e.g. ABCDEEFF). From this graph, it should be apparent that entropy provides a much more accurate picture of how well a given key partitions a file than the percentage of unique keys does.



Minimum and Maximum Entropy vs. % Unique Keys
For Files of 100 Records(x) and 5000 Records(o)

GRAPH 1

6.5 Title Key Development Based on Entropy Measures

Entropy is a powerful tool for measuring precisely the effectiveness of different keys and key elements down to the single character level. For example, individual character positions can be tested to find those positions having the best dispersion. Then combinations of these "best" positions can be tested to see how they "correlate" (entropy measures being the best approximation to the well-known correlation coefficient available for non-scaled variables). In this fashion a key for titles (or any other record element) can be more rationally developed than with the largely heuristic methodologies used in the past.

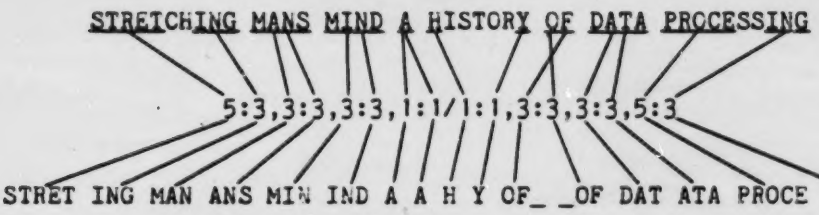
To apply this measure to title keys, we generated three large character position keys and calculated the entropy for each position and for several combinations of positions. These tests were performed on 5,046 titles from an OCLC test tape. Several keys tested approached the maximum obtainable entropy of 0.9636 (the file contained 1,257 duplicate records).

The first key tested was 20 characters long and consisted of the first two and last two characters of the first, second, third, next to last, and last words of the title. The motivation for this key was based on one result of Fokker and Lynch (that keys formed from the front and back of words were more discriminating than those derived from the front only) and the general perception that the last words of titles often carry as much information as the front.

Entropy was calculated for each of the 20 positions, for each of the ten pairs (first two characters of the first word, etc.), for the five pairs formed from the first and last letter of each word, for a triple made up of the first character from each of the first three words, and for a ten character key formed from the first and last character of each of the five words. This entire sequence of tests was repeated on the same key with vowels suppressed. That is, a similar 20 character key consisting of the first and last two consonants from the first, second, etc. words of the title was formed and tested. The consonants only keys in general had a lower entropy than the keys based on the complete alphabet. This generally poorer performance of the consonants-only keys presumably resulted from the reduction in the variety of characters available.

The second key tested consisted of the first eight characters of the first five words of the title. Short words and short titles were padded with blanks to arrive at a fixed length 40 character key. This general key was used to test "OCLC-type" keys. It was also useful for documenting the effect of short words and short titles on the effectiveness of these keys. A consonants-only version was also tested. One interesting finding from this test was the jump in entropy at the third character of title words.

The third key tested was based on the previous two and provided some unanticipated results. It was a 5:3,3:3,3:3,1:1/1:1,3:3,3:3,5:3 key.

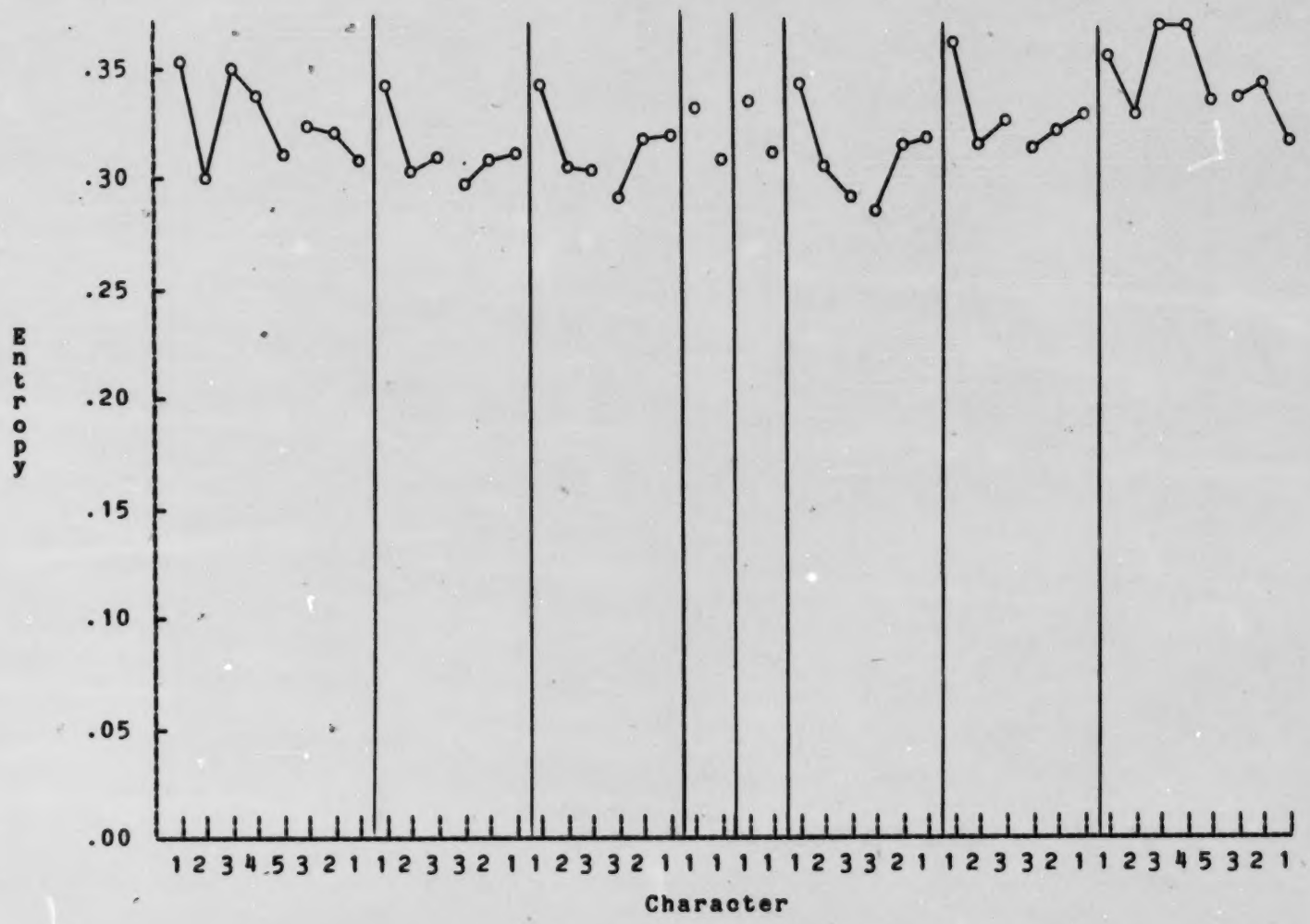

STRETCHING MANS MIND A HISTORY OF DATA PROCESSING
5:3,3:3,3:3,1:1/1:1,3:3,3:3,5:3
STRET ING MAN ANS MIN IND A A H Y OF_ _OF DAT ATA PROCE ING

Among the character positions not previously tested were the two highest positional entropies found. The graph is also notable for its symmetry (see Graph no. 2). A consonants-only version was not tested due to the poor performance of consonant keys in the previous tests.

Once all the high entropy (good discriminator) positions had been found, the next step was to select those 6 to 10 positions whose combination would produce the best possible partition (i.e. the smallest number of groups of matching keys). The final size of the key would depend on how effective it was. If the best six character key still produced an unacceptablely high number of large matching groups then a larger key would be required. The obvious selection method was to combine the positions with the highest individual entropies. This was done for a six character key (1;3:/1:,1;3-4) and it was tested on the 5,046 record OCLC test file. The key matched 24 title pairs, only two of which were false drops (i.e. the titles did not actually match).

Merely selecting the high entropy positions however, is not necessarily the best method for finding a "best" key since two high entropy positions might be highly correlated, resulting in a relatively low entropy for the pairing. In the extreme case, two perfectly correlated positions would have the same entropy and the pairing would have that same entropy. Thus developing a key from these individual positions required calculating the entropy of combinations of positions. Ideally, to find the best N character key, one would calculate the entropy for every N combination of the 44 positions. However this would involve 44!/(44-N)! expensive calculations. For example, if N=8 this approach would require 7.146×10^{12} separate program executions.

So, as an heuristic approach, calculations were done only for pairs of positions from the 12 positions (out of the 44) having the highest individual entropy scores. All 66 possible pairings of the 12 positions had their "pairwise" entropies calculated. Then for each six and eight combination of the twelve positions the sum of all possible pairwise entropies was calculated. The top keys and their scores (sums) were:



Positional Entropies For a 44 Character Key
(5:3,3:3,3:3,1:1/1:1,3:3,3:3,5:3)

GRAPH 2

Size	Key	"Score" (sum of pairwise entropies)
6	1;3,,1:/1:,;3-4	10.27
	1;3:/1:,1:,;3-4	10.26
	1;3:,1:/1:,;3-4	10.25
	1;3:/1:,1;3-4	10.25
	1:,1:,1:/1:,;3-4	10.23
8	1;3:,1:,1:/1:,1;3-4	19.01
	1;3:,1:/1:,1:,1;3-4	19.00
	1;3:,,1:/1:,1:,1;3-4	18.98
	1;3:,1:,1:/1:,;3-4:2L	18.96
	1;3-4:,,1:/1:,1;3-4	18.95

Table no. 9 Candidate Title Keys

Testing beyond this point (all three-combinations, four combinations, or pairings for the top 16, etc.) was not attempted because of the processing costs and the limited potential for improvement. These keys perform quite well.

6.6 Addition of Dates to the Title Key

However, it was realized that even an optimum title key (one that distinguished every different title) would still produce large groupings of matching keys for "popular" titles such as Proceedings, Collected Works, etc. So at this point it was decided to add an additional element to the first pass key. The date element was chosen as an element that was short, amenable to fixed length keys, distinctive, and also not likely to be subject to variant cataloging practices. Thus, in the next tests of title keys, they were combined with the last two digits of the first date in the imprint. Two such keys have been tested with a sample file of 19,291 LC MARC records and performed beyond the level where entropy measures are useful discriminators (i.e. entropies for these keys were greater than 0.99 and the differences between them too small to be meaningful). Thus, while the methodology of positional entropy has not produced "the optimum key for titles", several near optimum alternatives have been isolated for closer testing. At this stage of key development, the number of unique keys a given key format produced became the more useful measure of key performance. This, plus manual inspection of the resulting record pairs were used as the last steps in defining a first pass title-date key.

6.7 Test Results for Candidate "Title-Date" (TD) Keys Using the LC MARC File

One six character key and one eight character key were tested on the sample file of 19,292 LC MARC records. The six character key that was chosen from the keys listed above was the one that eliminated the two false drops (mis-matches) resulting from the test of the six character key with the best individual entropies against the OCLC test file reported above. This was the 1:,1:,1:/1:,;3-4 key.

In the 19,292 record file it found 87 matches and since MARC files do not contain duplicate records, these were false drops. Sixty-five of these varied in title and 22 of the 87 varied in the other areas such as publisher, paging, edition, or author. Since the false drop ratio is the only significant measure of performance, it was used for analyzing key effectiveness. This key found false drops (mis-matches) for 0.45% of the file.

The next key tested was the eight character key from the pairwise testing that also was the best eight character key when only individual entropies were considered. This was the 1;3:,1:/1:,1;3-4: key. It found 39 false drop pairs and the same 22 other variant pairs for a false drop (mis-match) ratio of 0.20%. By inspection and comparison, a variant eight character key: 1;3:,1:,1:/1:,1:,;3-4 (which unfortunately did not show up in the best pair entropy keys) should lower the false drop pairs down to 25 or 0.13% of the file.

In later tests, it was determined that this key performed poorly for one word titles. That is, most of the false drops found had one word titles. A slightly altered key was the one finally used for the Title-date key for IUCS: 1;3:,1:,1:/1:,1:,;4:2L. This key has the advantage that up to four characters are selected from one word titles: the first, third, fourth, and second from last.

An additional enhancement of the first pass process was the production of multiple keys for records having multiple dates, one key for each year. By doing this, two editions of a work having imprints with the date sub fields "1971" and "1975 [c1971]" could be paired and determined to be duplicates if they passed all later tests. This capability is in accordance with our definition of a duplicate (see Section no. 2).

This title-date key used for IUCS -- first and third characters of the first title word, first character of second and third words, first character of second and third words from the end of the title, and fourth and second from last characters from the last word -- is not intended to stand alone, but is only to be used as a first pass or initial screen. Finer screening is provided in the second pass and the false drops (mismatches) that pass the TD key screen should be detected in the second pass.

7. Developing a Second Pass Screen

The matching (locating duplicates) procedure is divided into two passes. The first works with the entirity of both files to be matched and proceeds by extracting and sorting character keys from the records in the file to be added and comparing them with character keys from the existing union catalog file. The details of this character key, the title-date key, were described in the previous section. The second pass does more detailed testing on the records passing this first pass screen (i.e. records having matching title-date keys).

The second stage (or second pass) consists of three more detailed matching tests (screens) for those records that were matched in the first stage. There are matching tests for authors, titles (Harrison-Hamming test), and paging. In the second pass, pairs of records are compared so if N>2 keys were grouped in the first pass (e.g. Proceedings, 1974), then N(N-1)/2 pairings of records must be compared in the second pass for that grouping.

7.1 Author Matching

Since determination of main entry is often one of the most difficult cataloging decisions, it tends to be one of the most variable in practice. For this reason, records are being considered potential matches if they match on any main/added entry combination and even, in some cases, if there is no author match.

Because of the other tests being done, comparing the first five characters of author entries has been quite sufficient for determining matching authors. An exception has been made for the small percentage (see Table no. 5) of corporate and conference main/added entries. For these, the initial words have been tested against a set of stop words, eliminating all initial words that are found (see Appendix C). This is designed to find the first key word in the entry and to suppress the discrepancies between "U. S.", "U.S.", and "United States" and between "Gt. Brit." and "Great Britain". All punctuation is suppressed as well. Since corporate and conference authors occur infrequently, this extra processing is not expensive.

As the authorship match is being done, markers are set to indicate two things about the pairing. One indicates whether either (or both) records has no author field (main or added entry). The other indicates whether or not the match found was between added entries. These indicators are used in conjunction with the paging match algorithm (described below) to determine whether a pairing should be visually examined.

Even if there is no match between authors, the pairing is still tested for title match and paging match. If it matches exactly on these two tests, then it is printed out for visual examination in the "Authorship failure" file. A large majority of these pairings (exact title and paging match but no authorship matching) have been duplicates in our tests.

7.2 Full Title Matching Using Harrison Keys and Hamming Distance

Harrison keys are bit strings derived from character strings. The Hamming distance between two bit strings is the number of positions in which the two bit strings differ and is a useful measure of their similarity.

The Hamming distance test was used as the second test of the second pass to determine duplication of titles. It permits a closer look at title matching than the title key above does, but does not involve character by character comparisons. It gives an indication of the extent of the difference, if any, between two titles. It allows us to take title pairs which differ only slightly, possibly due to typographic errors or small variations in the recording of the titles, and treat them as duplicates whereas they would be classified as non-duplicates on the basis of exact title matching.

The Hamming distance between two titles is equal to the difference between the Harrison keys [Harrison] which are derived from each title. The Harrison key is a bit string* in which the individual bits are "set" (changed from zero to one) according to certain characteristics of the title. The length of the bit string, i.e., the number of bits, can vary. We have tested both two-word Harrison keys, 72 bits long, and four-word Harrison keys, 144 bits long. The two-word key appears better for our purposes. A Harrison key is derived from a title as follows:

1. All letters are capitalized.
2. All punctuation and spacing is removed so that the title becomes one continuous character string. Differences in punctuation and spacing, therefore, will not affect the matching of two titles.
3. The resultant character string is divided into overlapping trigrams, i.e., each set of three consecutive characters is considered separately. ABCDE would consist of three separate trigrams: ABC, BCD and CDE.
4. Each trigram is used to determine a bit position in the Harrison key as follows: Each character's representation in the computer is a 7-bit bit string. The 3-bit strings representing a trigram are treated as a 21 bit binary number. The remainder when this number is divided by 71 determines which bit in the Harrison key that trigram "sets" or "turns on". For example:

Trigram	Computer Representation	Decimal value	Remainder
ABC	1000001 1000010 1000011	1,073,475	26
BCD	1000010 1000011 1000100	1,089,988	67
CDE	1000011 1000100 1000101	1,089,988	37
EDC	1000101 1000100 1000011	1,139,267	1

The remainder determines the bit position that is set to 1 in the Harrison key for the string containing the trigram. As can be seen from the last two examples, this mapping is position dependent: permutations of the same three letters set different bits.

*A bit is the elemental unit of information storage and can assume only the value of one or zero. A bit string is a series of such "characters".

The same trigram will always set the same bit and, thus, the same titles will always generate the same Harrison key. It is also possible, however, for different trigrams to set the same bit and, thus, for different titles to produce the same Harrison key. The probability that identical Harrison keys represent identical titles is, nevertheless, quite high. Different Harrison keys will always represent different title strings.

The Hamming distance between two Harrison keys is equal to the number of positions in the two keys in which the bits are set differently. Thus, in two Harrison keys, if the first, third, fifth, and seventh bits of the first key are set to one and the first, third, fifth, and eighth bits of the second key are set to one, the difference between the two keys, i.e., the Hamming distance, is two. The keys differed in the seventh and eighth bits.

As an example of this process, consider the two titles The Stranger and The Stronger. They would first be turned into the two character strings THESTRANGER and THESTRONGER. The following two sets of trigrams would then be generated:

THE HES EST STR TRA RAN ANG NGE GER
THE HES EST STR TRO RON ONG NGE GER

Note that the first group contains three trigrams which are not present in the second group and that the second group contains three trigrams which are not present in the first group. Each trigram would then be used to set a bit in the Harrison key associated with that title. The first group of trigrams would set nine bits, as would the second group, but three of the trigrams from the first title would very likely set bits not set by any trigrams from the second title and three of the trigrams from the second title would very likely set bits not set by any of the trigrams from the first title. Considering only the bits set to one, all others remaining at zero, the resultant Harrison keys could be visualized as follows:

First title:
0000 1000000000000000 10 1000000000000000 100000 1 1 1000 1000000000000000000000

Second title:
0000000000000000000 10 1000000000000000 10000000 100 1 10 100000 1000000000000

Position where keys differ:
0000 1000000000000000000000000000000000000000000 1 1000 100 100000 1000000000000

Hamming distance = 6

The two titles may represent two different books or they may represent the same book with one of the titles containing a typographic error. With our Hamming distance cutoff set at eight these two records would be treated as potential duplicates and tested further for paging matches.

If our Hamming distance cutoff were set at six or lower, the two records would be considered non-duplicates and would not be checked further. The maximum Hamming distance between two titles which differ only in one character, as in this example, is 6. Because of the possibility of two different trigrams setting the same bit, however, the

distance may be less than 6.

The greater the difference between the two titles the greater the Hamming distance is likely to be. The Hamming distance test thus gives us an indication of whether or not two title strings differ plus some indication of the extent of that difference.

7.3 Paging Matching

The final test in the pass 2 screen is for paging differences. The results of the paging test are combined with the results from the authorship and title matching to determine whether a particular pair of records represent duplicates, potential duplicates (requiring examination), or non-duplicates.

The basic procedure for determining the paging for an individual record consists of isolating the paging subfield in the collation field and then scanning it for arabic numbers. The algorithm continues scanning until the next number found is less than the last. (The scanner returns a zero if no number is found.) The page numbers selected by this algorithm are the underlined numbers in the following examples:

iii, 151 p.
xviii,5-469 p.
[2], 27 p.
xii, 148, [2] p.
382 p. [16] p. of plates
A-K, 366 p.
2v. (xxxi, 999p.)
[24] p.
[37] leaves (3 fold.)
xii, 35, 35 p.
vii, [35] p., 12 leaves of plates
v. (no paging found, return 0)
[51]p., [159] p. of illus. (part col.)
8v. in 5

Testing showed that the paging difference by itself was insufficient to separate duplicate pairings from non-duplicate pairings since many duplicates had large paging differences. One additional paging characteristic that seemed relevant was the size of the smaller of the two paging values. That is, given the two pagings, which is smaller and how large is it? For example, some duplicate pairs had large paging differences because in one record the item was recorded as a two volume set while in the other, the total paging was recorded, e.g. "2 V." as opposed to "999 P." The size of the smaller paging also helped in identifying non-monographic records such as filmstrips, map collections, etc., that may be found in "non-serial" collections of bibliographic records. Using these two variables in combination enabled the algorithm to successfully separate the duplicates from non-duplicates for many of the record pairs. Further testing was required to analyze some combinations of paging difference and the smaller paging value. The following table gives the twelve categories arrived at, their probable cause, and the action to be taken in each case. The actions are explained below:

Paging Diff	smaller pg.	Probable Cause	Action
0	0	cip or open vol.:cip or open vol.	Pr/AE/NA
0	1-10	vol.:vol.	Pr/AE/NA
0	>10	----	Dup
1-5	0	vol.:cip or open vol.	Pr/AE/NA
1-5	1-10	vol.:vol.	Examine
1-5	>10	----	Dup
6-15	0	vol.:open vol.	Pr/AE/NA
6-15	1-10	----	Examine
6-15	>10	----	Pr/AE/NA/HD
>15	0	paging:cip or open vol.	Pr/AE/NA
>15	1-10	paging:vol.	Examine
>15	>10	----	Not Dup

Table no. 10
Paging Situations Found in Pairs of Records that are Potential Duplicates
(cip=cataloging in publication record; vol.=volume; :=versus)

The four basic actions that can be taken with respect to record pairs after their paging test are (in order of increasing "degree of match"):

1. Reject as non-duplicates (Not Dup)
2. Examine (Examine)
3. Check for other suspicious conditions and examine if found (Pr,AE,NA,HD)
4. Accept as duplicates (Dup)

Actions 1,2, and 4 are the basic outputs of the algorithm. Action 3 needs further explanation. Pr,AE,NA, and HD all refer to characteristics of a pairing or the records in it that can be determined during the author and title testing and that make it "suspect" (i.e. records and pairings with these characteristics tend to be questionable matches). Pr means the title began with the character string "PROCEED" or "ABSTRACT" (meaning that the record is probably a procceedings of a conference or abstracts from a conference). AE means the author match was an Added Entry-Added Entry match (as opposed to a Main-Main or Main-Added match). NA means that one or both records had no authorship given (main or added). HD means the Hamming distance was greater than four (the title matching test requires that the Hamming distance be less than or equal to eight). If the paging conditions require any of these conditions to be checked and one or more of them occur, then the record pair is marked for examination.

8. IUCS Process for Generating a Union Catalog

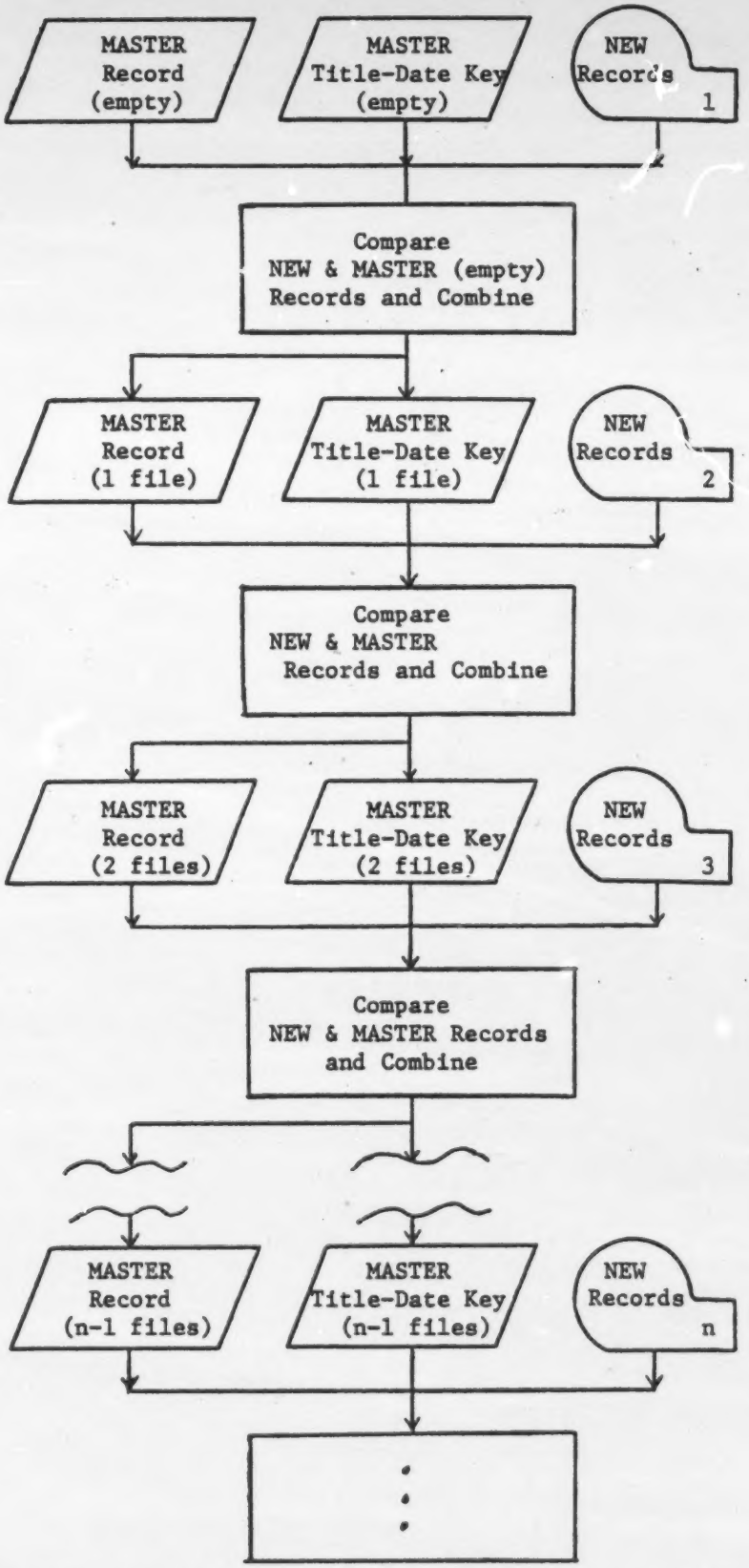
The process of generating a union catalog file involves the addition of multiple files to the first file in a system (called the MASTER file). The steps described in this section involve the IUCS procedure for adding one NEW file to the MASTER file. Adding additional files would involve repeating this process. (See the following flow charts.)

Starting conditions: The MASTER record file and the NEW record file both contain bibliographic records in machine-readable form. MASTER is the union catalog at this point and NEW contains the records to be added to it. The MASTER Title-Date key file contains key records consisting of a file number, a record number, the title-date key and a pointer to the associated record in the record file (see page no. 63). The following steps combine MASTER and NEW record files into an updated MASTER record file with the associated updated TD key file. The nine steps in the overall process include pre-processing, pass 1, pass 2, and post-processing:

(1) Translate the NEW file to the IUCS record format (see page no. 62);
(2) Create a Title-Date (TD) key file for the NEW record file, and sort this NEW TD key file on the key;
(3) Compare the MASTER & NEW TD key files outputing a file consisting of groups of matching TD keys;
(4) Input the matching TD key groups to the PASS2 program which accesses the associated records from MASTER and NEW record files for further testing. The three output files from PASS2 consist of sets of pointers to record pairs (both cross file and internal to NEW). One file contains pointers to pairs that match, the second contains pointers to record pairs that may match (Examine), and a third contains keys to records that are presumed to match but may not (Authorship Failure).
(5) Use the file of pointers to confirmed matches in order to automatically mark duplicates. For internal duplicates in the NEW record file, the duplicate without an LC card no. (if there is one) will be marked for later deletion;
(6) Use the questionable match pair pointers (Examine and Authorship Failure) to generate printouts of the record pairs for visual inspection. The file of pointers can then be used to selectively mark matching records (Examine) or erase marks (Authorship Failure);
(7) Sort NEW TD key file on record no;
(8) The CONCAT program concatenates all records in NEW that have not been marked as duplicates to the MASTER record file. At the same time, CONCAT reads the TD key file associated with NEW, discarding the keys of duplicate records and updating the pointers for non-duplicate records that are being concatenated;
(9) Re-sort the NEW TD key file on the TD key and merge it with the MASTER TD key file.

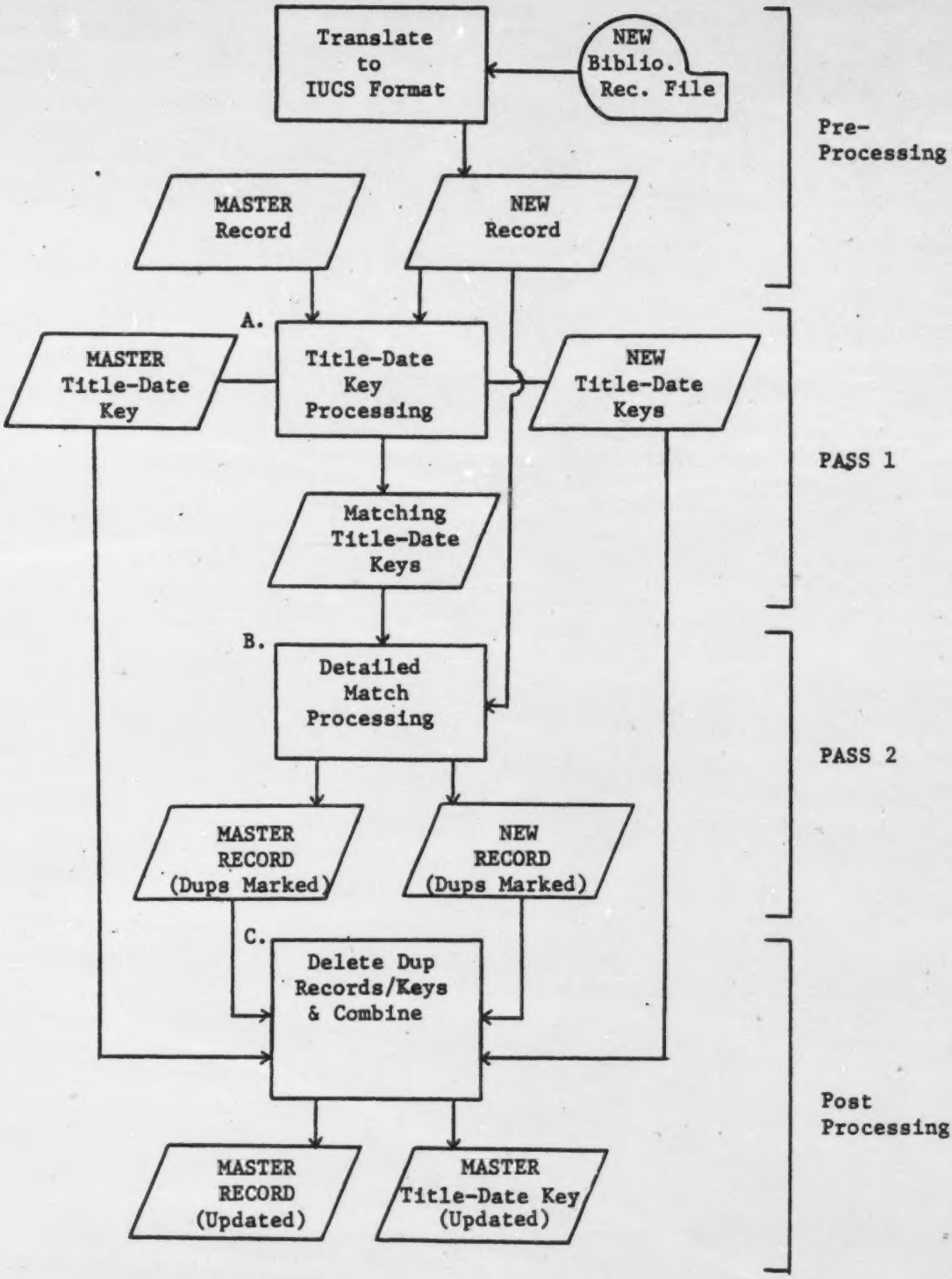
The first step constitutes the pre-processing, steps two and three represent pass 1, steps four through seven are pass 2 and steps eight and nine are the post-processing steps.

OVERVIEW OF INTERATIVE COMBINING PROCESS

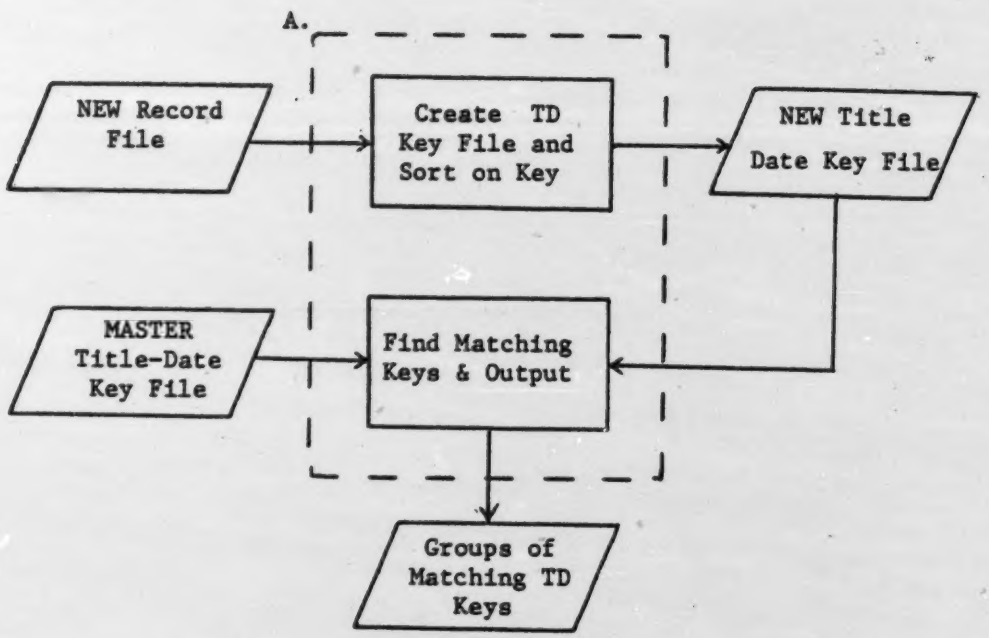

MASTER Record (empty)
MASTER Title-Date Key (empty)
NEW Records 1
Compare NEW & MASTER (empty) Records and Combine
MASTER Record (1 file)
MASTER Title-Date Key (1 file)
NEW Records 2
Compare NEW & MASTER Records and Combine
MASTER Record (2 files)
MASTER Title-Date Key (2 files)
NEW Records 3
Compare NEW & MASTER Records and Combine
MASTER Record (n-1 files)
MASTER Title-Date Key (n-1 files)
NEW Records n

* This process is the same for initializing and updating the MASTER file.

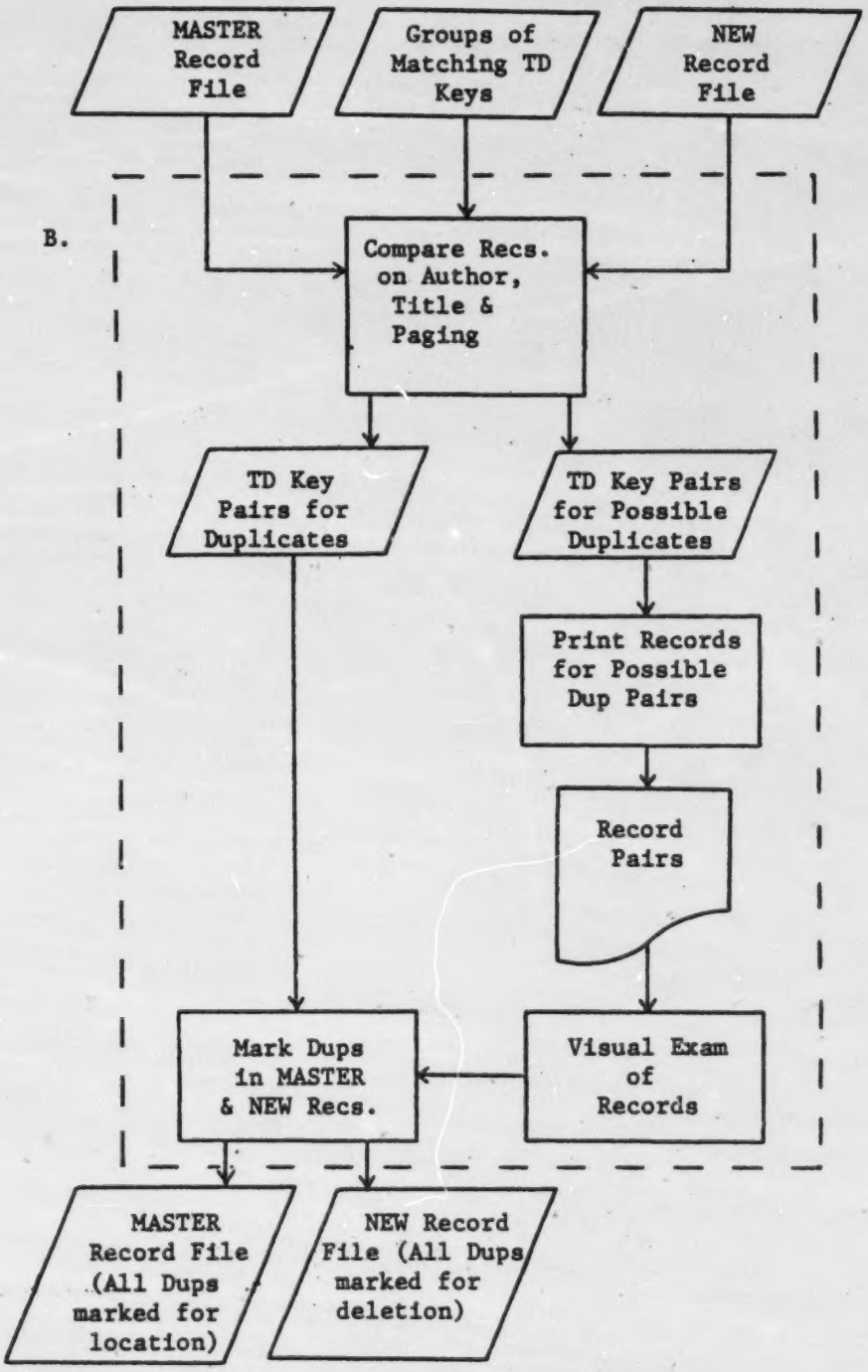
COMPARE & COMBINE PROCESS



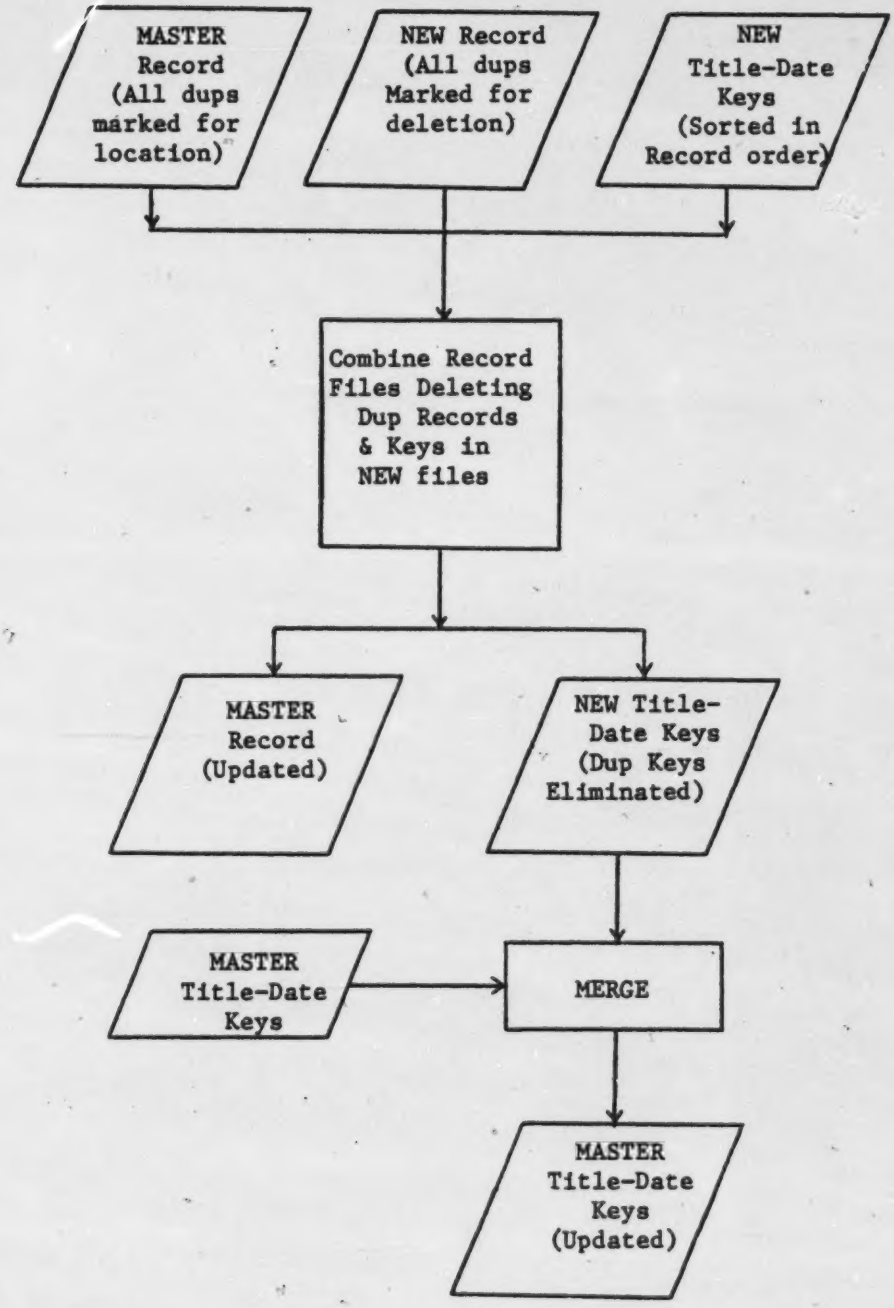
A. TITLE-DATE KEY PROCESSING
(Create - Compare)


A.
NEW Record File
Create TD Key File and Sort on Key
NEW Title Date Key File
MASTER Title-Date Key File
Find Matching Keys & Output
Groups of Matching TD Keys

B. DETAILED MATCH PROCESSING
(Compare - Mark)


MASTER Record File
Groups of Matching TD Keys
NEW Record File
B.
Compare Recs. on Author, Title & Paging
TD Key Pairs for Duplicates
TD Key Pairs for Possible Duplicates
Print Records for Possible Dup Pairs
Record Pairs
Mark Dups in MASTER & NEW Recs.
Visual Exam of Records
MASTER Record File (All Dups marked for location)
NEW Record File (All Dups marked for deletion)

C. POST PROCESSING (Delete - Combine)


MASTER Record (All dups marked for location)
NEW Record (All dups Marked for deletion)
NEW Title-Date Keys (Sorted in Record order)
Combine Record Files Deleting Dup Records & Keys in NEW files
MASTER Record (Updated)
NEW Title-Date Keys (Dup Keys Eliminated)
MASTER Title-Date Keys
MERGE
MASTER Title-Date Keys (Updated)

9. Full Scale Test of IUCS

9.1 Summary of Test Results -- Including Extrapolation to a Million Record Base File

The IUCS process discussed above (Section 8) was implemented on three large files -- 73,552 records from OCLC for library X, 57,728 records from NW, and 22,857 records from UC. All records were for items entered into the respective libraries during the 1974-75 time period. The largest sample file used in this project, the OCLC file, was chosen to be the base or MASTER file to which the other files would be added. The first step was to identify internal duplicates in the OCLC file. The IUCS found 702 duplicate records in this file. (By a linear extrapolation, this indicates 9,544 duplicates in a collection of 1,000,000 titles, using the IUCS definition of duplicate bibliographic records (see Section no. 2).) The IUCS required manual examination of 32 record pairs. (If this were a million title file, the number of pairs to be manually examined would be 435.)

The first file merged with this was the Northwestern file. It contained 57,728 records. IUCS found 14,946 duplicate records of which 728 were internal duplicates. 14,218 Northwestern records matched records in the OCLC file. (Extrapolating on the basis of a million record OCLC file, this represents a 784,860 record file in which the IUCS would find 203,203 duplicate records representing 9,898 internal duplicates and 193,305 "cross-file" duplicates for a file overlap of 24.63% of the Northwestern file.) In the test, 167 record pairs required manual examination (2271 record pairs would require manual examination in the extrapolated example).

This overlap figure of 24.63% is based on a simple linear extrapolation of our test results. However, it is expected that the overlap (as a percentage of the file size) would grow with larger files. We attempted to approximate the overlap behavior of large files by taking realistic size sample files from large machine-readable collections; to ensure overlap of records we took a time slice -- records added to the test collections during 1974-1975. We can only guess what the actual overlap figures would be for larger files.

After combining the OCLC and NW files the resulting MASTER file (or union catalog) had 115,641 records (or 1,572,235 in the extrapolation). To this new MASTER file we next added a somewhat smaller sample file from the University of Chicago. It contained 22,857 records (which represents a sample of the cataloging output of UC for 1974-75). The IUCS found 42 internal duplicates and 5,130 records that matched records in the MASTER file. Seventy-three record pairs required examination. (These figures extrapolate to adding a file of 310,760 records having 571 internal duplicates and 69,747 records already in the union catalog. 992 record pairs would require examination.) Overlap between the MASTER file, which at this point contained OCLC and NW combined, and the UC file was 22.44% of the UC file. This figure is slightly lower than the Northwestern figure of 24.63% and the difference is probably the result of a smaller file size, possibly tempered by the larger size of the MASTER file. (If the overlap were 60%, a linear extrapolation would predict that 2,652 record pairs would require examination.)

For the purposes of this project one figure of interest that is related to overlap is the number of record pairs that require manual examination. The record examination process is not an onerous burden.

In the process of testing IUCS, approximately 1,500 record pairs were printed out and were examined by a library science student in less than two days time. A scheme for matching records that is less effective than IUCS would require more manual effort and be proportionately more expensive as the cost of manual matching is considerably more expensive than computer processing.

9.2 Test Results for Merging OCLC:NW:UC Files

Test results for the merging of OCLC, NW, and UC are presented in tabular form. The tables present the significant data related to the merging of these files in conformance with the IUCS process as described in Section 8. Tables 11-19 contain the figures given above and additional data relating to the performance of the IUCS. Several of the figures require some explanation. the "No. of TD keys" is always larger than the number of records since records with two dates in the date subfield resulted in two keys being generated. "Pass 1 Results" gives the details of the first pass. Sets of matching TD keys can be any size and these columns give the details of the resulting matches. For a fuller description of "Entropy", see Section 6 on development of a first pass key. The key for the largest set remained the same through all tests: P0 CG73. This key most probably represents works having the title "Proceedings" and the date "1973". This is exactly what one might expect to be the most popular title-date combination in 1974-75 files from research libraries.

The next table, "PASS2 Results", gives the figures for the three classes of output from the PASS2 program: "Duplicate Key Pairs," for record pairs passing all PASS2 tests; "Examine Key Pairs" for record pairs classified as failing the PASS2 tests but having a medium probability of being duplicates; and "Authorship Failure Key Pairs" which are presumed to represent duplicate record pairs (they are marked as duplicates by IUCS) but have a small probability of not being duplicates. See section 7 which describes the matching algorithm for details on how membership in these three classes is determined by the PASS2 program. This table also includes the results of the manual examination of the records in the "Examine" and "Authorship Failure" classes. Finally, the totals from the three classes plus manual examination are given. The "Total Duplicate Records Eliminated" figure is smaller than the "Total Duplicate Key Pairs" figure due to two factors: (1) Multiple keys for some duplicate records and (2) the multiway matching done in PASS2 for key groupings larger than two. For instance, if three matching TD keys represent records which all match under the PASS2 requirements, then the result is 3 duplicate key pairs (1-2,1-3,2-3) but only two duplicate records eliminated (2 and 3).

1. Generation of MASTER (OCLC vs. OCLC) to Find Internal Duplicates

	No. of Records	No. of TD Keys
OCLC	73,552	76,611

Table no. 11 Starting Conditions

Set Size	No. of Sets	No. of Records	No. of Pairs
2	904	1,808	904
3	42	126	126
4	6	24	36
5	2	10	20
6	1	6	15
7	1	7	21
8	3	24	84
9	1	9	36
14	1	14	91
24	1	24	276
	962	2,052	1,609

Entropy: 0.9981

Table no. 12 Pass 1 Results (Matching TD Keys)

Duplicate Key Pairs	736
Examine Key Pairs	26
Duplicates	12
Authorship Failure Key Pairs	6
Non-Duplicates	1
Total Duplicate Key Pairs	747
Total Record Pairs Examined	32
Total Duplicate Records Eliminated	702
Total Records Added	72,850
Total Records in Union Catalog	72,850

Table no. 13 Pass 2 Results

2. Addition of NEW file (NW) to MASTER file (OCLC)

	No. of Records	No. of TD Keys
OCLC (dups removed)	72,850	75,858
Northwestern	57,737	60,673
	130,587	136,531

Table no. 14 Starting Conditions

Set Size	No. of Sets	No. of Records	No. of Pairs
2	14,605	29,210	14,605
3	587	1,761	1,761
4	45	180	270
5	10	50	100
6	4	24	60
7	4	28	84
9	2	18	72
10	2	20	90
13	1	13	78
14	1	14	91
17	1	17	136
35	1	35	595
	15,263	31,370	17,942

Entropy: 0.9858

Table no. 15 Pass 1 Results (Matching TD Keys)

Duplicate Key Pairs	15,316
Internal	(1,097)
Cross-file	(14,219)
Examine Key Pairs	111
Duplicates	99
Internal	(10)
Crossfile	(89)
Authorship Failure Key Pairs	56
Non-Duplicates (Internal)	3
Total Duplicate Key Pairs	15,412
Internal	(1,104)
Crossfile	(14,308)
Total Record Pairs Examined	167
Total Duplicate Records Eliminated	14,946
Internal	(728)
External	(14,218)
Total Records Added	42,791
Total Records in Union Catalog	115,641

Overlap 24.63% (of Northwestern File)

Table no. 16 Pass2 Results

3. Addition of NEW file (UC) to MASTER file (OCLC+NW)

	No. of Records	No. of TD Keys
Master (OCLC+NW)	115,641	121,056
UC	22,857	23,794
	138,518	144,850

Table no. 17 Starting Conditions

Set Size	No. of Sets	No. of Records	No. of Pairs
2	5,843	11,686	5,843
3	115	345	345
4	21	84	126
5	7	35	70
6	6	36	90
7	4	28	84
8	1	8	28
9	2	18	72
10	2	20	90
11	2	22	110
13	1	13	78
14	1	14	91
18	2	36	306
21	1	21	210
27	1	27	351
44	1	44	946
	6,010	12,437	8,840

Entropy: 0.9947

Table no. 18 Pass 1 Results (Matching TD Keys)

Duplicate Key Pairs	5,160
Internal	(59)
Cross-file	(5,101)
Examine Key Pairs	48
Duplicates	36
Internal	(3)
Crossfile	(33)
Authorship Failure Key Pairs	25
Non-Duplicates	1
Total Duplicate Key Pairs	5,195
Internal	(62)
Crossfile	(5,133)
Total Record Pairs Examined	73
Total Duplicate Records Eliminated	5,172
Internal	(42)
External	(5,130)
Total Records Added	17,685
Total Records in Union Catalog	133,326

Overlap 22.44% (of UC File)

Table no. 19 Pass 2 Results

10. Testing IUCS for Mismatches and Missed Matches

There are two types of errors that a record matching program can make. It may omit matching two records that do relate to the same document (hereinafter called "missed matches") or it may erroneously match two records that do not relate to the same document (hereinafter called "mismatches").

Missed matches cause unnecessary records to be left in the MASTER file, making it larger, and may cause location information to be missed (if the searcher finds one record without finding the other). Mismatches are a more serious error, causing information to be permanently lost, since a unique record will be eliminated from the MASTER file.

10.1 Mismatches

Based on a manual examination of 3,686 record pairs we found two erroneous matches, one of which represented erronous cataloging and one of which represented a defect in IUCS. Thus 99.946% of the matches determined by IUCS are correct. The two mismatched pairs are bibliographically related works -- two different translations of the same work and the student and teacher versions of the same text.

The 3,686 records examined included all of the OCLC (MASTER file) internal duplicates (747), plus 2055 of the duplicates found when NW was added to OCLC, plus 884 of the duplicates found when the UC file was added. By making a minor change in IUCS (checking the authorship statement or the 700 fields for indication of translation), IUCS matching effectiveness would increase to 99.973%.

The two mismatched pairs are:

100 EURIPIDES.
245 ALCESTIS, $C BY EURIPIDES [WITH] A TRANSLATION WITH COMMENTARY BY CHARLES ROWAN BEYE.
260 ENGLEWOOD CLIFFS, N.J. $B PRENTICE-HALL $C [1974]
300 XXVII, 116 P. $C 21 CM.
490 GREEK DRAMA SERIES
700 BEYE, CHARLES ROWAN $E TR.

100 EURIPIDES.
240 ALCESTIS. ENGLISH
245 ALCESTIS / $C EURIPIDES ; TRANSLATED BY WILLIAM ARROWSMITH.
300 XI, 122 P. ; $C 25 CM.
490 THE GREEK TRAGEDY IN NEW TRANSLATIONS
700 ARROWSMITH, WILLIAM, $D 1924-

110 SCHOOL MATHEMATICS STUDY GROUP.
245 MATHEMATICS FOR HIGH SCHOOL: GEOMETRY. [STUDENT'S TEXT] $C
PROPARED UNDER THE SUPERVISION OF THE PANEL ON SAMPLE TEXTBOOKS OF
THE SCHOOL MATHEMATICS STUDY GROUP: FRANK B. ALLEN [AND OTHERS]
250 (PRELIMINARY ED.)
260 [NEW HAVEN? $B YALE UNIVERSITY, $C C1959]
300 2 V. $B ILLUS.

110 SCHOOL MATHEMATICS STUDY GROUP.
245 MATHEMATICS FOR HIGH SCHOOL: GEOMETRY. $C COMMENTARY FOR TEACHERS.
PREPARED UNDER THE SUPERVISION OF THE PANEL ON SAMPLE TEXTBOOKS OF
THE SCHOOL MATHEMATICS STUDY GROUP: FRANK B. ALLEN [AND OTHERS]
250 (PRELIMINARY ED.)
260 [NEW HAVEN? $B YALE UNIVERSITY, $C C1959]
300 2 V. $B ILLUS.

The first of these is a match according to the IUCS duplicate definition, but since these two documents would not be considered duplicates by most people, they point up a defect in our definition. It should be pointed out that these two items had to have the same year of publication and similar pagings to be matched. Thus very few variant translations would be mismatched by IUCS.

The second example shows a cataloging error: "COMMENTARY FOR TEACHERS" should be part of the subtitle and if it had been the records would not have matched.

10.2 Analysis of Missed Matches

In order to test the effectiveness of the IUCS, we tried to locate matching records using different first pass keys, namely, the LC card number and the ISBN. The resulting grouping were then tested with the PASS2 program. Those that passed PASS2 were tested to see if they would have been paired by matching title-date (TD) keys. Failing this test indicated a failure of the TD key. Those pairs that passed the first pass key but failed the PASS2 screen were examined manually to determine whether they were non-duplicate pairs (possibly indicating a failure of the alternate key -- see Section no. 4) or duplicate pairs (indicating a failure of the PASS2 program).

Cross file matching was done for the OCLC and Northwestern files using LC card numbers and ISBNs. The total records in the test files for OCLC and NW used in this test and the numbers of records containing LC card nos. and/or ISBNs is as follows:

	OCLC	NW
Total records	73552	57728
Records having 010 or 020 field	46441	51022
Records having LC card no.	42001	50508
Records having good ISBN no.	26164	24337
Records having both	22557	23895

Table no. 20 LC and ISBN Occurrences in OCLC and NW

Matching OCLC and NW files using a variety of first pass matching criteria resulted in our being able to estimate how many true matches were missed by the IUCS. A discussion of the types of missed matches follows. Reasons for missed matches fall into four categories: (1) variation in titles, (2) variation in date data, (3) variation in paging data, and (4) variation in author data.

10.2.1 Variation in Titles

Eleven different types of title errors relating to content were categorized on the basis of record pairs failing IUCS tests:

Variation in Subfield Delimiters
Subtitle Missing
Words Added
Word Substitution (Including Number Variations)
German Abbreviations
Author's Name Included
Encoding Variations
Typographic Errors
Bracketted Expressions
Word Delimiter Variations
Spacing of Initials and Abbreviations

Title -- Subfield Delimiters

This was the most serious category of missed duplicates, accounting for 35% of the total. While disappionting, it is probably unavoidable, since although the title usually gets recorded consistently, decisions on what constitutes the title, subtitle, and author statement can vary considerably. By using the title and subtitle fields ($a and $b), title checking is not hampered by variant decisions as to where the subtitle starts. However variation in the location of $c (author statement) does cause problems and is a common error. For example:

THE ALPHA BETA STORY; $C AN ILLUSTRATED HISTORY OF A LEADING WESTERN FOOD RETAILER, BY ESTHER R. CRAMER vs.
THE ALPHA BETA STORY; $B AN ILLUSTRATED HISTORY OF A LEADING WESTERN FOOD RETAILER, $C BY ESTHER R. CRAMER.

One possible solution would be to take the Title portion of the Title-Date key from the fronts of titles only and use the Harrison keys to determine whether one title was the substring of another or not. (This can be done by using different logical operations on the Harrison keys). This procedure would increase the work for PASS2 since in the testing of the TD key, keys taken only from the front did considerably worse than keys taken from both ends. And this method still would not save short titles where there was variation in the recording of the author statement.

Title -- Subtitle Missing

This was the second most troublesome error. Several of these were conference proceedings such as:

BIOLOGY IN PEST AND DISEASE CONTROL vs.
BIOLOGY IN PEST AND DISEASE CONTROL $E THE 13TH SYMPOSIUM OF THE BIRTISH ECOLOGICAL SOCIETY, OXFORD, 4-7 JANUARY 1972.

Others were simply long or questionable subtitles that got left out:

THE MESSAGE IN THE BOTTLE / vs.
THE MESSAGE IN THE BOTTLE : $B HOW QUEER MAN IS, HOW QUEER LANGUAGE IS, AND WHAT ONE HAS TO DO WITH THE OTHER /

Title -- Words Missing

Generally, the words added were phrases in the subtitle or at the end of the title. For example:

HIGH VOLTAGE ELECTRON MICROSCOPY: $B PROCEEDINGS vs.
HIGH VOLTAGE ELECTRON MICROSCOPY: $B PROCEEDINGS OF THE THIRD INTERNATIONAL CONFERENCE

WHEN I SAY NO I FEEL GUILTY : $B HOW TO COPE--USING SKILLS OF SYSTEMATIC ASSERTIVE THERAPY. vs.
"WHEN I SAY NO I FEEL GUILTY" : $B HOW TO COPE WITH OTHER PEOPLE, USING THE VERBAL SKILLS OF SYSTEMATIC ASSERTIVE THERAPY.

These failed the Harrison key test in PASS2 and the first one would have also failed on TD keys. The same techniques proposed above might work for these categories to a limited extent.

Title -- Word Substitions (Including Number Variations)

The most popular of the word substitutions was AND vs. &, but we also had AND vs. UND, OF vs. FOR, and AU vs. DU. Number variants included Roman vs. Arabic and such things as 20'S vs. TWENTIES as in this example:

THE THEATRICAL 20'S / $C ALLEN CHURCHILL. vs.
THE THEATRICAL TWENTIES / $C ALLEN CHURCHILL.

These errors in general passed the Harrison key test but failed the TD key test when they occurred in the first or last three words of the title.

Title -- German Abbreviations

This might also be thought of as subcategory of word substitutions. Setting the Hamming distance cutoff at eight helps in catching many of these but some cannot be included without raising the cutoff and matching large numbers of non-duplicates. This example exceeded the eight cutoff in the Hamming distance test:

SOZIOLINGUISTIK. $B EXEMPLAR. U. KRIT. DARST. IHRER THEORIE, EMPIRIE U. ANWENDG. $C MIT KOMMENTIERTER BIBLIOGRAPHIE. vs.
SOZIOLINGUISTIK. $B EXEMPLARISCHE UND KRITISCHE DARSTELLUNG IHRER THEORIE, EMPIRIE UND ANWENDUNG. $C MIT KOMMENTIERTER BIBLIOGRAPHIE.

In general, abbreviations are less of a problem for the TD key since much of the key comes from first letters of words, but the cases where it failed were those where the end of the last word was abbreviated. Two examples are SCHICKSALSDICHTG. vs. SCHICKSALSDICHTUNG and BEITR. : BEITRAGE.

Title -- Author's Name Included

Unfortunately, there are no clear cut guidelines as to when the author's name can or should be included in the title statement. Thus we have the following examples of variant titles:

T. S. ELIOT'S THE COCKTAIL PARTY vs.
THE COCKTAIL PARTY

AESCHYLUS: THE ORESTEIA vs.
THE ORESTEIA

GRAHM SUTHERLAND. NEUE WERKE. RECENT WORK. vs.
NEUE WERKE. RECENT WORK.

ARTHUR SYMONS : POETRY & PROSE / vs.
POETRY & PROSE

Title -- Encoding Variations

There are three basic "micro" errors (errors at the individual character level) that can effect the functioning of the TD key: encoding differences, typographic errors, and spacing differences that effect the word boundary determinations. The last of these was the most troubling (see below). Encoding differences occur when different computer systems handle special characters differently. The most troublesome in these tests was the French "OE" for which OCLC had a special character and Northwestern used the two characters "O" and "E". For example:

AIM^E C^ESAIRE, L'HOMME ET L'^UVRE vs.
AIM^E C^ESAIRE, L'HOMME ET L'OEUVRE

In our system all special characters were translated to the character "^". These characters were then suppressed when keys were formed. This procedure avoided difficulties when diacritical marks were included by one library but not another, but did lead to the problem described above.

Titles -- Typographic Errors

these were much less troublesome that originally expected. Variations in word delimiters (see below) were more of a problem for the TD key. Most examples we found involved foreign languages and using "L" for "1" in numbers (see also year and paging problems below). One example in English was ENGLIGHTENMENT vs. ENLIGHTENMENT as the first word of a title.

Title -- Bracketted expressions

There is a trade-off involved between suppressing bracketted expressions in titles and accepting them as part of the title. It was felt that since the bracketted information is usually supplied by the cataloger and would thus be subject to variation in practice, it would be better to suppress it. This was indeed the case in a smaller test with control numbers where both methods were tried. When bracketted information was suppressed from the title before forming the Harrison key, ten missed duplicate pairs were "recovered" and two pairs, previously classified as duplicates, were "lost" (not matched) for a net gain of eight. As can be seen from this larger test, only five duplicate pairs were missed due to this type of error. In the first item of this exmaple, the bracket was accidentally closed with a parenthesis, causing everthing after the bracket to be suppressed.

PAN-AFRICAN LANGUAGE IN THE WESTERN HEMISPHERE; $B PALWH [P^LWH): A RE-DEFINITION OF BLACK DIALECT AS A LANGUAGE AND THE CULTURE OF BLACK DIALECT vs.
PAN-AFRICAN LANGUAGE IN THE WESTERN HEMISPHERE; $B PALWH [PAELWH]: A RE-DEFINITION OF BLACK DIALECT AS A LANGUAGE AND THE CULTURE OF BLACK DIALECT

Title -- Word Delimiter Variations

One problem faced by all character key generating programs is that of word delimiting or what to do with dashes, slashes and compound words. For the title-date key it only comes into effect when it has an effect on the word determination aspect of the title-date key, that is, when it occurs in the first or last three words. for example the two titles:

RIBBIN', JIVIN', AND PLAYIN' THE DOZENS: $B THE UNRECOGNIZED DILEMMA OF INNER-CITY SCHOOLS vs.
RIBBIN', JIVIN', AND PLAYIN' THE DOZENS: $B THE UNRECOGNIZED DILEMMA OF INNER CITY SCHOOLS

result in two different title keys: RBJAQIOL and RBJAICOL (derived from the underlined letters), since inner-city is treated as a single word.

Other examples from this test:

SELF PORTRAIT vs. SELF-PORTRAIT
SUPER MARKETS vs. SUPERMARKETS
TWELFTH-CENTURY vs. TWELFTH CENTURY
PARENT POWER, CHILD POWER vs. PARENT POWER/CHILD POWER

Titles -- Spacing of Initials and Abbreviations

Another case of unclear typographic practices for catalogers, this relates not only to people's initials but also U.S. vs. U. S., WASHINGTON D.C. vs. WASHINGTON D. C., SHEFFIELD, U.K. vs. SHEFFIELD, U. K., etc.

The actual numbers for these title errors are as follows:

Variation in Subfield Delimiters	122
Subtitle Missing	49
Words Added	39
Word Substitutions (Including Number Variations)	20
German Abbreviations	12
Author's Name Included	9
Encoding Variations	5
Typographic Errors	8
Bracketted Expressions	5
Word Delimiter Variations	21
Spacing of Initials and Abbreviations	16
Miscellaneous	16
	322

10.2.2 Variation in Date Data

Two major causes found for year failing to match in the TD key were different years and the use of "L" instead of "1".

Different Years

This was a very frustrating error to find since most everyone would agree that two records with different years of publication represent different works. Yet, we found several records that had the same control numbers and matched in every field except the year. While most of these had brackets around the year in one or both of the records (indicating that the date was found somewhere besides the title page), some did not. For example:

```
010  74007759
020  0394491025
100  KENNER, HUGH.
245  A HOMEMADE WORLD;  $B THE AMERICAN MODERNIST WRITERS.
250  [1ST ED.]
260  NEW YORK, $B KNOPF;  [DISTRIBUTED BY RANDOM HOUSE] $C 1974
300  XVIII, 221, IX P.  $C 22 CM.

010  74007759
020  0394491025
100  KENNER, HUGH.
245  A HOMEMADE WORLD;  $B THE AMERICAN MODERNIST WRITERS.
250  [1ST ED.]
260  NEW YORK, $B KNOPF, $C 1975
300  XVIII, 221, IX P.  $C 22 CM.
```

"L" Instead of "1"

This problem appears almost everywhere where numbers are involved in these records. Here, this problem might have been solved by looking for "L" or "1" when locating the year, but this approach has not been tried.

The totals for these problems are:

Different years	
Brackets for one or both	27
No Brackets	8
"L" instead of "1"	7
	49

10.2.3 Variation in Paging Data

Mismatches caused by paging errors were categorized as typographic error in recording paging, large variation in paging, and variation in recording paging.

Typographical error in recording paging

Large paging differences were considered to be typographical errors when two digits were exchanged (108 vs. 180) or only one out of 3 digits differed (338 vs. 388) or lower case l had been used for a numeric 1 (43L vs. 431). These latter cases were apparent since all letters were capitalized in our files. While the first two situations are nearly impossible to deal with, the last could be attacked by searching the paging field for L's having an adjacent digit. However, this approach presents fresh difficulties since that 43L in the above example might mean 43 leaves. This approach has not been tried.

Large variation in paging

These may actually be typos, but they did not fall into any of the above categories. Most of these had no discernible reason for the variation. For example:

```
010  74010677
020  0070457387
100  NABOKOV, VLADIMIR VLADIMIROVICH, $D 1899-
245  LOOK AT THE HARLEQUINS!  $C BY VLADIMIR NABOKOV.
260  NEW YORK, $B MCGRAW-HILL $C [1974]
300  214 P.  $C 22 CM.

010  74010677
020  0070457387
100  NABOKOV, VLADIMIR VLADIMIROVICH, $D 1899-
245  LOOK AT THE HARLEQUINS $C BY VLADIMIR NABOKOV.
260  NEW YORK, $B MCGRAW-HILL $C [1974]
300  253 P.  $C 22 CM.
```

Some varied in height also which might indicate a hardbound vs. paperback variation.

Variation in recording of paging

These in general are caused by variations in cataloging practice, for example:

16 P., 24 LEAVES OF PLATES vs. 16 P., [48] P. OF PLATES

[48] P. vs. [24] P., [24] P. OF PLATES, LEAF OF PLATE TIPPED IN

While these seem to suggest possible changes in the paging algorithms (see Section no. 7.3) that would cause these to match such as ignoring bracketted paging or adding various pagings together, numerous counterexamples where these methods would fail were found in the development of the current algorithm. In other words, no one algorithm can decipher all the vagaries of paging and these are simply the cases where the current algorithm fails. From previous testing, we believe these are the least likely possibilities among all the variations possible in the recording of paging in bibliographic records.

The counts for these types of errors were:

Typographic Error in Recording Paging	11
Large Variation in Paging	7
Variation in Recording of Paging	5
	23

10.2.4 Variation in Author Data

Three causes for missing matches because of authorship were found. They are: encoding variations, spacing, and no authorship match. There was only one example of each of these categories. The number of missed duplicates due to authors would be much higher than the three recorded here if it were not for the fact that record pairs for which there is no authorship match are allowed to match if the Hamming distance and paging difference are zero and the smaller paging is greater than ten. Thus the three author failures noted here had additional problems above and beyond no authorship match. There were 20 pairs that were duplicates and only failed on authorship matching.

Author -- Variation in Encoding

Certain letters or letter combinations in foreign names have a special code in the ASCII 8-bit character set. If one library uses the special code while another uses the nearest equivalent letter(s) from the standard code then there can be problems with the matching. If, when the records are converted to a standard format, the special characters were converted to characters from the standard code, this would not be a problem. We did not do this however. All unusual characters were converted to "^" so we had some matching failures. For example: DAN^, SVEN vs. DANQ, SVEN

Author -- Spacing

This might be considered to be a common mistake even though we only have one example: DU_BOIS, WILLIAM EDWARD BURGHARDT, $D 1868- vs. DUBOIS, WILLIAM EDWARD BURGHARDT, $D 1868-1963. This type of error

could be simply overcome by suppressing blanks in the authorship field before forming the five character matching key.

Author -- No Authorship Match

The last category of author failure would seem to be an unusual one indeed; two records for the same item with no matches in authorship. The one example from this comparison is due to an unusual mistake (two main entries) in one of the records. The machine algorithm arbitrarily takes the last occurence of a field when there are multiple occurrences of a single occurrence field such as author-title, and collation so the underlined portions in the following two records were compared:

100 TATTERSALL, BRUCE.
245 STUBBS AND WEDGWOOD; $B UNIQUE ALLIANCE BETWEEN ARTIST AND POTTER, $C WITH AN INTROD. BY BASIL TAYLOR.
260 [LONDON] $B TATE GALLERY $C [1974]
300 119 P. $B ILLUS. (PART COL.) $C 26 CM.

100 TATTERSALL, BRUCE.
110 TATE GALLERY, LONDON.
245 STUBBS & WEDGEWOOD; $B AN EXHIBITION.
245 STUBBS AND WEDGWOOD : $B UNIQUE ALLIANCE BETWEEN ARTIST AND POTTER / $C BRUCE TATTERSALL ; WITH AN INTROD. BY BASIL TAYLOR.
260 LONDON, $B TATE GALLERY, $C 1974.
260 [LONDON] : $B TATE GALLERY ; $A PARIS ; $A MILAN : $B DISTRIBUTED BY IDEA BOOKS, $C 1974.
300 119 P. $B ILL. ; $C 25 CM.
700 STUBBS, GEORGE, $D 1724-1806.
710 TATE GALLERY, LONDON.

10.2.5 Summary

The total IUCS missed matches are as follows:

Title	279	81.1%
Year	49	12.3%
Paging	23	5.8%
Author	3	0.8%
Total	397	

Of the 11,823 duplicates found, 397 were duplicates that IUCS would not have located.

11. Testing the Effectiveness of IUCS vs. LC/ISBN

In order to test the effectiveness of the IUCS title-date (TD) we compared the matching effectiveness (capability for identifying duplicate records that truly are duplicates) of the IUCS TD key vs. the LC card number and the ISBN. We also tested the matching effectiveness of combinations of these. Each of these keys, codes, or combinations was tested by matching files containing a known number of duplicates. In order to determine the true number of duplicates, we matched two files using all techniques including IUCS PASS2 and considered the sum of all pairs of duplicates found -- regardless of technique -- plus 1.37% to be 100% of duplicates. The 1.37% is the percentage of duplicates missed by the IUCS PASS2 screen (see Section no. 10.2 for discussion).

The files used were the OCLC file of 72,850 records and the NW file of 57,737 records. the total number of pairs of duplicates found using the IUCS TD key plus LC card number, plus ISBN was 14,441. By adding 1.37% or 202 to the 14,441 we determined that the total number of pairs of duplicates to be found in these files is 14,643 (100%). The results of matching the two files using the various keys, codes and combinations of same are presented in Table No. 21. Neither LC nor ISBN alone was as effective as the IUCS TD key, nor was the use of LC and/or ISBN as effective as the use of TD and/or LC or ISBN or both.

	Matching Key/Code	Number of Duplicates Matched	Matching Effectiveness (%)
single key/code	LC	10,968	74.90
	ISBN	8,285	56.58
	IUCS-TD	14,219	97.10
combination key/code	LC/ISBN	11,823	80.74
	IUCS-TD/LC	14,425	98.51
	IUCS-TD/ISBN	14,375	98.17
	IUCS-TD/LC/ISBN	14,441	98.62

Table no. 21 Matching Effectiveness of IUCS-TD vs. LC vs. ISBN

12. Computer Costs for IUCS Processing

Estimating computer processing costs for a different installation based on cost figures for IUCS is difficult due to the fact that different installations have different charging schemes, processor speeds, etc. This problem is compounded in this case since the work was done on a non-IBM machine and was written in a non-standard programming language. However, by giving a cost range based on the Day-Night range of rates for this system, hopefully some idea of the magnitude of these costs can be communicated. The following estimates are based on charges of $70/hour for cpu and 1.75/hour for a kilobyte of core during the day and $7/hour for cpu and $.18/hour for a kilobyte of core after midnight. Costs for IUCS are categorized by the four steps in one IUCS iteration in which a NEW file is added to a MASTER file: Pre-processing, Pass 1, Pass 2, and Post-processing. The computer programs employed in the various IUCS processes are referred to in this section using program names. A full description of the functions of each program is given in Appendix B.

12.1 Pre-Processing Costs

Pre-processing consists of one program to translate NEW files with non-standard record formats and data to a standard format containing a standard subset of bibliographic data elements (for instance, all subject information can be eliminated to save space). This was the most expensive single step in the entire system in our testing, and yet it would have to be done for any machine-readable union catalog, regardless of the methodology for identifying duplicates.The daytime rates ranged from $131.17 to $144.89 (per 100,000 records) while the night rates were $13.23-$14.66.

12.2 Pass 1 Processing Costs

Pass 1 consists of two major processes: generation of keys from the new file (TTLRDR) and comparison with the existing keys (COMEXT). TTLRDR is the most expensive here since it, like the translating programs above, involves serially reading an entire record file. However, the amount of processing performed is less than that of the translating programs. TTLRDR costs would range from $52.59 to $47.68 for daytime processing (per 100,000 records) to $5.30-$4.80 for after midnight.

COMEXT is less expensive since it involves reading key files which are much smaller than record files. Also, there is no range of costs (ranging over different record files) since all key files are essentially the same in terms of the difficulty of processing. The processing cost for COMEXT per 100,000 records (generating 104,503 keys) is $2.52 daytime, $.25 night. While the translation and TTLRDR costs covered above depend only on the size of the NEW file, COMEXT costs depend on the size of the MASTER plus NEW files since it compares all existing keys with the NEW keys. Thus if a 200,000 record file was being added to a million record file, the actual COMEXT cost would be 12

times what is listed here. (See below for an extended example of costing.)

12.3 Pass 2 Processing Costs

The actual costs for the processing in Pass 2 depend on the amount of overlap between the NEW and MASTER files. PASS2 costs are extrapolated on the basis of the number of matching keys input to it. This works out to $8.96 per 10,000 keys daytime or $.90 at night. The number of keys, based on our tests, is estimated to be 2.2 times the number of actual duplicate pairs found. After running PASS2, some record pairs need to be printed for examination. At $.05 per page, 1,000 record pairs would cost $20.67 to print.

Marking duplicates (TDMARK) is the next step. This cost depends on how many duplicate key pairs are found by the PASS2 program. The average cost, based on our processing, was $3.92 day, $.40 night per 10,000 records marked (5,000 duplicate pairs). The actual costs, again, depend on the amount of overlap.

12.4 Post-Processing Costs

After marking the duplicates, the final step is to add the non-duplicate records from the NEW file to the MASTER file. The CONCAT program essentially reads and writes records so the extrapolation of its costs is based on the number of records read or written. The estimated total cost per 100,000 records read or written is $4.93 daytime and $.50 at night.

The figures discussed above were extrapolated from actual runs that merged three large sample files from OCLC, Northwestern, and UC. The actual costs for processing these three files are given in table no. 22.

	OCLC	NW	UC
Pre-processing	$9.75	$7.65	$3.28
Pass 1			
TTLRDR	$3.85	$2.80	$1.08
COMEXT	$0.20	$0.33	$0.36
Pass 2			
PASS2	$0.22	$2.80	$1.12
TDMARK	$0.05	$1.18	$0.45
Printing (Examine pairs)	$0.70	$3.50	$1.55
Post-processing			
CONCAT	$0.68	$0.51	$0.26
Total	$15.45	$18.77	$8.10

Table no. 22 Costs for Full Scale Tests on IUCS

12.5 Estimated Cost of Adding Files to a MASTER File of One Million Records

In order to provide a rough basis for estimating the cost of generating a union catalog of full scale large files, we have estimated the costs of adding a 200,000 record NEW file to a million record MASTER file with 25%, 50%, and 75% overlap. All estimates are based on figures developed in the merging of the 1974-75 files from OCLC, NW, and UC. Costs are based on the daytime and nighttime rates on the U of I, CSL's DECsystem 10 computer.

		Day	Night	
Pre-processing				
Translation		$289.78	$29.32	
Pass 1				
TTLRDR		105.18	10.60	
COMEXT		30.24	3.00	
Percentage overlap		25%	50%	75%
Records matching		50,000	100,000	150,000
Keys matching		110,000	220,000	330,000
Pass 2				
PASS2	day	$98.56	$197.12	$295.68
	night	$9.90	$19.80	$29.70
TDMARK	day	$39.20	$78.40	$117.60
	night	$4.00	$8.00	$12.00
Printing costs (based on 1.193% of dups being examined)		$12.33	$24.66	$36.99
Post-processing				
Total records read or written to MASTER		350,000	300,000	250,000
CONCAT	day	$17.26	$14.79	$12.33
	night	$1.75	$1.50	$1.25
Total machine cost	day	$592.55	$769.49	$887.80
	night	$70.90	$96.88	$122.86

Table no. 23

Costs for Adding a 200,000 Record File to a Master File of 1,000,000 Records With Overlaps of 25, 50, and 75%

13. Benefit of Research Results and Future Needs

The feasibility study has proved the feasibility of automatically generating a union catalog from non-standardized machine-readable files inexpensively and with a high degree of accuracy. Most systems for matching records employ simple keys or codes such as LC card number or ISBN. The simple keys (e.g. OCLC or INOVAR) generally match multiple records (5-10 or more) and require human examination to determine which of the candidate matches are true matches.

If, for example, a file of one million records were matched with a file of 200,000 records and the resulting overlap were 50% or 100,000, in a search key system that produced an average of five hits or potential matches for each input record, the number of records to be manually examined would be 600,000. In IUCS, approximately 1.1% of the duplicates identified require manual verification thus in the match of 200,000 vs. 1,000,000 with 50% overlap, the number of records to be manually examined would be 2,200.

Codes such as LC or ISBN are reasonably effective when they are present but a significant portion of the records in most large libraries would not contain these codes. In addition to the fact that the codes are not present on all records is the fact that an unacceptably high number of errors are present in the codes as they are recorded in multiple libraries. Transcription errors and the lack of a check digit account for LC errors and ISBN errors are largely due to the fact that publishers have been reusing ISBNs. IUCS provides a better matching technique because IUCS keys and codes are present for 100% of the records. In addition, IUCS is much more sensitive and provides a higher match effectiveness.

Reasons why IUCS provides effective matching are:

Codes are present on 100% of records to be matched. The IUCS keys and codes are generated from the records and thus are present on all records brought into the system.

Overcomes typographical and spelling errors. By allowing some latitude in Hamming distance matches, a certain number of typographical errors can be overcome.

Overcomes differences in format. By disregarding differences in upper and lower case letters, differences in spacing and certain kinds of punctuation and by conversion of all records to IUCS format, IUCS overcomes the possibility of missing matches due to differing formats.

Overcomes transcription errors. By allowing some latitude in Hamming distance matches, a certain number of transcription errors can be overcome.

Overcomes variation in cataloging. A large number of matches that would be missed because of variation in cataloging are able to be identified because IUCS matches authors (main and added), IUCS suppresses bracketted information in titles, accomodates different ways of counting and representing paging, accomodates German abbreviations in titles, matches any dates in the collation, and ignores publisher, size and edition.

Overcomes differences in corporate entry. IUCS can accomodate some of the different representations in difficult corporate entries such as U.S. vs. United States vs. US or Gt.Brit. vs. Great Britain.

IUCS can be used: (1) to generate a union catalog, on an

institution-wide, state-wide, or regional basis; (2) to update a union catalog -- the same procedure used for generating the catalog is used for updating it; (3) to augment the records of a library and (4) to create a machine-readable file for a library; (5) to provide a capability for inexpensive searching to identify records for interlibrary loan; (6) to provide a basis for subject searching by bit screen matching of search terms with the Harrison keys generated from titles; and (7) to provide a means of identifying duplicate retrieval results (identical records) retrieved from different data bases in on-line or batch information retreival systems.

The study was a research study to determine the feasibility of automatically generating a union catalog. The feasibility has been demonstrated. A few modifications could be made to increase the effectiveness of the screens but the limitations of time and money prevented this. If the results are to be exploited for the State of Illinois, it would be necessary to make minor modifications in the systems and to program the system in a higher level language for a popular computer system. Further work would be required in reprogramming and documentation in order to generate a transportable "turn-key" system.

Appendix A
Technical Specifications

Hardware

The computer used for this project was the Coordinated Science Lab's DECsystem-10 computer with a KI processor and 336K words of core (1680K bytes). Peripherals include:

- two 800 b.p.i. tape drives
- three 50 megabyte disks (RP03) (2 user mountable)
- two 80 megabyte disks (RP04)

The rates on the CSL DEC-10 are $7.00-$70.00 per hour of cpu time, $.18-$1.75 per hour for a kiloword of core, $.35-$3.50 per hour connect time, $.02 per 128 words per month disk storage, and $.05 per page of printed output. The variable charges vary with time of day.

Software

The programs were written primarily in SAIL, a dialect of ALGOL 60 developed at the Stanford Artificial Intelligence Laboratory. Some routines were written in MACRO-10, the DECsystem-10 assembly language.

Bibliographic Record Structures

The records received from different institutions were translated into a standard format for use in this project. The overall structure resembles that of LC MARC transport record in that each record consists of a leader, a variable length directory, and the variable length fields. This basic format was modified to meet the requirements of this project and take advantage of the characteristics of the DECsystem-10.

The leader consisted of three 36-bit words. The first word contained the number of variable length fields in the left half-word and the total length of the variable length fields in the right half-word. The second word contained a record number in the right half-word and a file number in the left half-word. The third word was reserved to indicate holdings. Each bit in this word indicates whether or not an item represented by the record was held at the library represented by that bit position. To illustrate, if the rightmost four bits represented libraries A,B,C and D, respectively, then the record for an item held only by A would have the following bit pattern in this HOLDINGS word:

000000 000000 000000 000000 000000 001000

while a book held by all four would have the following bit pattern in its record:

000000 000000 000000 000000 000000 001111

In this fashion, holdings information can be held as compactly as possible and can be recorded without changing the size of the record.

This three word leader is followed by a directory which consists of a variable number of computer words, one for each variable length field. Each of these "entries" in the directory contains the tag number for the field in the left half-word and the length of that field in the right half-word.

The variable length fields are stored in one continuous string with no delimiters (this string is actually stored as an integer array that can be converted to a string when needed for processing -- this speeds up input/output processing). Thus individual fields can be accessed by reading through the directory keeping track of the total length of the fields skipped until the desired tag number is found. Then a substring function picks out the desired field for processing.

Record Key Format

Title-Date keys consisted of four computer words each. The first word of each key contains the same information as the second word in the record leader, a physical file number in the left half-word and a record number in the right half. The second and third words of the keys contain the actual TD key stored in six-bit code. This allows up to 12 characters in the TD key (2 x 6), only ten of which have been utilized in this project. The fourth word contains a pointer to the start of the associated record. A typical key pair might look like this:

Word 1 file	rec. no.	Word 2 Word 3 key		Word 4 pointer
000001	10680	A ITMA HD	73	000007415327
000003	01369	A ITMA HD	73	000000938559

This represents record number 10,680 in file number one and record number 1,369 in file number three.

Appendix B
Major Processing Programs

OCLC, NW, UC (Format Conversion and Statistics)
Input: Tapes containing MARC format records supplied by the various institutions. 800 b.p.i. blocked (NW) and unblocked (OCLC,UC).
Output: A record file in the IUCS record format and a formatted field statistics file.
Description: A separate format conversion and statistics generating program was written for each of the files processed. The program reads the MARC format records from the relevent institution, uses the record indicators to decide which records to keep, gathers statistics on all fields and outputs pre-specified fields in IUCS formatted records. After reading a tape it calculates and formats the following data about each field found: Number of occurrences, percent of records containing the field, average size in characters, average size over entire file, maximum length, minimum length. In addition, this statistics file contains the number of records of each type (monograph, serial, error, etc.).
Auxillary programs: STRINT,NWCONF

TTLRDR (Title Reader)
Input: Any IUCS record file (specified at the terminal)
Output: A file of Title-Date keys (name specified at the terminal)
Description: This program reads through an IUCS record file isolating the title and passing it to the key forming routine in NEWTTL (New Title Key). TTLRDR takes the record number and calculates the starting address of each record. These data together with the key returned from NEWTTL are combined with the year of publication (two digits) extracted from the record. If the two dates of publication are found, two keys are created. This key is then written out to the output key file. This program also contains a testing module that can be requested from the terminal at run time. Using this, the key forming procedure in NEWTTL can be tested on titles input at the terminal.
Auxillary programs: NEWTTL (New Title Key), STRFIL (String Processing Procedures)

COMEXT (Compare and Extract)
Input: Any sorted IUCS Title-Date key file (specified from the terminal)
Output: 1) A file (name specified from terminal) consisting of groups of matching TD keys seperated by vertical tab characters. 2) Statistics on the number of groups by size and the entropy of the keys.
Description: The TD keys are read one at a time and compared to the preceeding key. If they match a grouping loop finds all the matching keys and writes them to Output adding an appropriate

increment to the entropy calculation and the group size statistics, otherwise, an increment for unique keys is added and the next key read and compared.

Auxillary programs: None

PASS2

Input: 1) A file consisting of groups of matching IUCS TD keys (file name specified from terminal at run time) 2) Two IUCS record files (name specified from the terminal at run time)

Output: Several files of TD key pairs, each one representing record pairs that meet certain criteria. The first four letters of all of these files is specified from the terminal at run time (see Section no. 7 for a fuller description of the pairings mentioned).

____AU.FAL The record pairs pointed to by these key pairs failed the authorship match and had either Hamming Distance > 0, or Paging Difference > 0, or Smaller Paging < 10.

____HD.FAL These keys point to record pairs with Hamming distance > 8.

____PG.FAL These records had an authorship match and Hamming distance < = 8 but had paging difference > 15 and smaller paging > 10.

____.EXM These are the record pairs to be examined - (see Section no. 7)

____AU.ODD These records had paging difference = 0, Hamming distance = 0, and Smaller paging > 10 but no authorship match. These record pointers are in the duplicate file but some may be non-duplicates. This file should be printed out and examined. RRERAS can be used to eliminate duplicate marks.

____DP.MTH These key pairs point to record pairs presumed to be duplicates by this program.

In addition, statistics indicating the number of key pairs that went to these files are printed out at the terminal at the end of execution.

Description: See the Section no. 7 for a detailed description of the logical operation of this program. The actual operation proceeds as follows:

1) Keys are read into an array until a group delimiter is found.
2) Each record pointed to by the keys in this array is randomly accessed and all authorship fields, the title field, and the collation field, are extracted and stored in arrays.
3) A double loop compares every record in this grouping with every other record as follows:
   A) The authors are matched
   B) The Hamming Distance is calculated
) The paging is compared

If a matching failure is determined at any point, the innerloop is exited. Key pairs are written to the appropriate Output files

as determined by these tests.

RRREAD (Random Record Reader)
Input: 1) A TD key file specified from the terminal. 2) One or two IUCS record files.
Output: A print file consisting of the records whose keys were in the TD key file.
Description: RRREAD randomly accesses the records pointed to by the TD keys, converts them to strings and formats them for Output, either to the terminal or to a file that can be printed offline. The basic use for this program in a production system is for producing the printouts of record pairs that need to be examined, records pointed to by the keys in ____.EXM and ____AU.ODD.
Auxillary programs: None.

TDMARK (Title-Date Marker), RRMARK (Random Record Marker), RRERAS (Random Record Eraser)
Input: 1) A TD key file (from PASS2) specified at the terminal. 2) Two IUCS record files: a master file with all internal duplicates already eliminated and a new file to be added to the master file. 3) The bit positions to be used to mark record pairs that match between the two files or one member of pairs that match internally in the new file.
Output: Master and new IUCS record files with duplicates marked.
Description: Each of these programs randomly accesses records in two files and alters the HOLDINGS word of the record header. TDMARK reads through the ____DP.MTH file (see PASS2 Description) and marks all records pointed to by the key pairs in it. RRMARK steps through the ____.EXM file marking only those records specified by an operator at a terminal. The ones to be marked have been determined from a printout of all the "examine" pairs. RRERAS does much the same for ____AU.ODD files only it erases the marks already made by TDMARK for pairs found to be non-duplicates.
Auxillary programs: None.

CONCAT (Concatenate)
Input: 1) Two IUCS record files ,specified from the terminal, one considered the master file and the other the new file to be added to it. 2) The TD key files associated with the new file, sorted in record no. order, also specified from the terminal.
Output: 1) A combined IUCS record file (master + new). 2) An updated TD key file for the new record file.
Description: This program reads through an IUCS record file skipping records marked as duplicates (both external and internal) and concatenating the rest to the master record file. At the same time, the TD key file associated with the new record file is also read. Keys associated with duplicate records are discarded. The remaining keys have their record number, file

number, and pointer fields updated to reflect the new location of the associated record.
Auxillary programs: None.

KEYPRT (Key Printer)
Input: TD key file.
Output: A TD key file in a print format.
Description: Since the TD keys are stored in a format that cannot be printed directly, this program is needed to allow checking key files by printing them out. It reads key files sequentially. The first and last words of the four word keys are converted to numbers and the middle two words are converted to alphanumerics.
Auxillary programs: None.

READER
Input: Any IUCS record file.
Output: Bibliographic records in a print format.
Description: Since IUCS record files are not stored in a printable format (the records are stored and read as integer arrays), this program is needed to check files. It reads IUCS record files sequentially and converts them to a print format that can either be directed to the terminal or to a file for off-line printing.
Auxillary programs: None.

Appendix C
Stopwords for Corporate/Conference Entries

It was recognized early in the development of the IUCS algorithms that the matching of corporate and conference entries could not be done as easily as the personal author matching, where we were simply comparing the first five characters. We still wanted a five character key so that personal and corporate entries could be treated alike, but certainly did not want authorship to be presumed to match when the first five characters (with punctuation suppressed) were "U S C" or "UNITE", etc. A striaghtforward and easily implemented procedure for overcoming this particular case is to skip over leading words found in a stopword list. For instance, the matching key for "U. S. Congress. Senate. Committee on Agriculture" becomes AGRIC if all the words preceeding it are on a stopword list.

Thus we set out to develop a stopword list for popular initial words in corporate/conference entries. The stopword list used by OCLC was selected as a starting list:

a,an,the	Dept.	New York
of,on	Department	Senate
American	Great Britain	Symposium
Association	Gt. Brit.	United Nations
Bureau	Gt.Brit.	U. N.
California	House	U.N.
Committee	India	United States
Conference	International	U. S.
Congress	National	U.S.
Council		

Table no. 24 OCLC Stopword List

Certain modifications had to be made to fit our system such as using all capital letters and suppressing punctuation. Also, two word phrases became two separate entries. To this, we added some words that seemed obvious corporate "buzz" words. Our second list looked like this:

A	COUNCIL	NEW
AMERICAN	DEPARTMENT	OF
AN	DEPT	OFFICE
ASSOCIATION	FEDERAL	ON
BOARD	FOR	REGIONAL
BRIT	GREAT	S
BRITAIN	GT	SEMINAR
BUREAU	GTBRIT	SENATE
CALIFORNIA	HOUSE	STATES
CANADA	INDIA	SYMPOSIUM
CANADIAN	INSTITUTE	THE
CENTER	INTERNATIONAL	U
COMMITTEE	N	UNITED
CONFERENCE	NATIONAL	US
CONGRESS	NATIONS	YORK

Table no. 25 Initial IUCS Stopword List

Since a longer list adds very little to the processing involved (doubling the list only adds one step, using binary search), and since we wanted to know whether we had missed anything, we ran a test on our two largest files to determine what the most frequent leading words in corporate entries (main and added) would be once the words in this list had been skipped. The following are the 20 most frequent, non-stoplisted, initial words in the NW file and the OCLC files together with their counts:

NW		OCLC	
FRANCE	224	SOCIETY	246
AKADEMIIA	141	OHIO	203
SOCIETY	141	UNIVERSITY	173
BRITISH	140	EDUCATIONAL	123
PARIS	140	ECONOMIC	117
UNIVERSITY	125	LABOR	113
RESEARCH	90	RESEARCH	100
EDUCATIONAL	89	TIMELIFE	97
SOUTH	81	HEALTH	95
LIBRARY	78	EUROPE	92
LONDON	75	SCIENCE	89
ECONOMIC	63	CHEMICAL	89
CENTRE	55	BRITISH	86
INSTITUT	55	EDUCATION	84
IEEE	54	SOCIAL	83
CHEMICAL	51	WORLD	81
INDIAN	50	PARIS	80
STATE	50	TEXAS	78
NORTH	48	LIBRARY	77
SOCIAL	48	STATE	74

Table no. 26 Most Frequent Initial Words after Stoplisting

From these we picked BRITISH, EDUCATIONAL, RESEARCH, SOCIETY, and

UNIVERSITY to be added to our Stopword List for Initial Words in Corporate and Conference Entries.

Appendix D

Bibliography

1. Ayres, F.H. "The Universal Standard Book Number (USBN): A New Method for the Construction of Control Numbers for Bibliographical Records." Program 8(3):166-173, July 1974.
2. Ayres, F.H. "The Universal Standard Book Number (USBN): Why, How and a Progress Report." Program 10(2):75-80, April 1976.
3. Barton, Ian J.; Lynch, Michael F.; Petrie, J. Howard; Snell, Michael J. "Variable-Length Character String Analyses of Three Data Bases, and Their Application for File Compression." In: Informatics 1: Proceedings of a Conference held by the Aslib Co-ordinate Indexing Group on 11-13 April 1973 at Durham University. London, Aslib, 1974.
4. Batty, David; Walker, Geraldene. "The TAFF (Title Abbreviation Formula Feasibility) Project in the College of Librarianship Wales." In: Cost Reduction for Special Libraries and Information Centers. Edited by Frank Slater. American Society for Information Science, Washington, D.C., 69-79, June 1973.
5. Beale, D. Diane; Lynch, Michael F. "An Evaluation of, and Improvement on, Ayres' Universal Standard Book Number." Program 9(2):35-45, April 1975.
6. Bookstein, Abraham. "A Hybrid Access Method for Bibliographic Records." Journal of Library Automation 7:97-103, 1974.
7. Bookstein, Abraham. "Statistical Behavior of Search Keys." Journal of Library Automation 6:109-116, 1973.
8. Chauveinc, M. "Automation of the Catalog at the University of Grenoble." Libri 21:191, 1971.
9. Coe, Mary Jordan. "Mechanization of Library Procedures in the Medium-Sized Medical Library: X Uniqueness of Compression Codes for Bibliographic Retrieval." Bulletin of the Medical Library Association 58: 587-597, 1970.
10. Cunningham, Jay L.; Scheiber, William D.; Shoffner, Ralph M. A Study of the Organization and Search of Bibliographic Holdings Records in On-Line Computer Systems. Phase I. Final Report. Berkeley, Institute of Library Research, University of California, 1969.
11. Dolby, James L. "An Algorithm for Variable-Length Proper-Name Compression." Journal of Library Automation 3:257-275, 1970.
12. Ehrhardt, F. "Computer Services from CAC Material in Germany." Presented at the EUSIDIC meeting in Luxembourg, October, 18, 1972.
13. Fokker, Dirk W.; Lynch, Michael F. "Application of the Variety-Generator Approach to Searches of Personal Names in Bibliographic Data Bases - Part 1. Microstructure of Personal Authors' Names. Part 2. Optimization of Key-Sets and Evaluation of Their Retrieval Efficiency." Journal of Library Automation 7:105-118, 201-213, 1974.
14. Guthrie, Gerry D.; Slifko, Steven D. "Analysis of Search Key Retrieval on a Large Bibliographic File." Journal of Library Automation 5:96-100, 1972.
15. Harrison, Malcolm C. "Implementation of the Substring Test by Hashing." Communications of the ACM 14:777-779, 1971.

16. Heaps, H.S. "Data Compression of Large Document Data Bases." Journal of Chemical Information and Computer Sciences 15:32-39, 1975.
17. Hilton, Howard J. A Code for the Unique Identification of Recorded Knowledge and Information. n.p., 1968.
18. INOVAR Corp. Author-Title Search Key, 1975?
19. Jolliffe, J.W. "Fingerprints and Search Codes." Catalogue & Index no.15:4-6, 1969.
20. Jolliffe, J.W. "Project LOC: Centralized Processing of Local Collections." In: Illinois, University of. Graduate School of Library Science. Clinic on Library Applications of Data Processing, 10th, Urbana, Illinois, 29 April-2 May 1973. Proceedings: Networking and Other Forms of Cooperation. Edited by F. Wilfrid Lancaster. University of Illinois, Graduate School of Library Science, Urbana, Illinois, 102-112, 1973.
21. Kilgour, Frederick G. "Retrieval of Single Entries from a Computerized Library Catalog File." In: Proceedings of the American Society for Information Science 5:133-136, 1968.
22. Kilgour, Frederick G.; Long, Philip L.; Leiderman, E.B. "Retrieval of Bibliographic Entries from a Name-Title Catalog by Use of Truncated Search Keys." Proceedings of the American Society for Information Science 7:79-82, 1970.
23. Kilgour, Frederick G.; Long, Philip L.; Liederman, E.B.; Landgraf, Alan L. "Title-Only Entries Retrieved by Use of Truncated Search Keys." Journal of Library Automation 4:207-210, 1971.
24. Kjell, B. "Performance of Kilgour's Truncation Algorithm in Files of Different Subjects." Journal of the American Society for Information Science 25:70-71, 1974.
25. Landgraf, Alan L.; Kilgour, Frederick G. "Catalog Records Retrieved by Personal Author Using Derived Search Keys." Journal of Library Automation 6:103-108, 1973.
26. Landgraf, Alan L.; Rastogi, Kunj B.; Long, Philip L. "Corporate Author Entry Records Retrieved by Use of Derived Truncated Search Keys." Journal of Library Automation 6:156-161, 1973.
27. Legard, Lawrence K.; Bourne, Charles P. "An Improved Title Word Search Key for Large Catalog Files." Journal of Library Automation 9:318-328, 1976.
28. Lipetz, Ben-Ami; Stangl, Peter; Taylor, Kathryn F. "Performance of Ruecking's Word-Compression Method When Applied to Machine Retrieval from a Library Catalog." Journal of Library Automation 2:266-271, 1969.
29. LOC Project. Computers and Early Books. London, Mansell, 1974.
30. Long, Philip L.; Kilgour, Frederick G. "Name-Title Entry Retrieval from a MARC File." Journal of Library Automation 4:211-212, 1971.
31. Long, Philip L.; Kilgour, Frederick G. "A Truncated Search Key Title Index." Journal of Library Automation 5:17-20, 1972.
32. Long, Philip L.; Rastogi, K.B.L.; Rush, James E.; Wyckoff, J.A. "Large On-Line Files of Bibliographic Data: An Efficient Design and a Mathematical Predictor of Retrieval Behavior." In: International Federation for Information Processing. IFIP Congress, 1971. Ljubljana, Yugoslavia, 23-28 August 1971. Proceedings North Holland Publishing Co., Amsterdam, The Netherlands, Booklet TA-3: Computer Software, 145-149, 1971.

33. Lowe, Thomas C. "Effectiveness of Retrieval Key Abbreviation Schemes." Journal of the American Society for Information Science 22:374-381, 1971.
34. Lynch, Michael F. "Variety Generation -- A Reinterpretation of Shannon's Mathematical Theory of Communication, and its Implications for Information Science." Journal of the American Society for Information Science 28:19-25, 1977.
35. Lynch, Michael F. "Creation of Bibliographic Search Codes for Hash-Addressing Using the Variety-Generator Method." Program 9(2):46-55, April 1975.
36. McCutchen, C.W. "Random Code Numbers for Universal Identification of Documents." American Documentation 16:91-96, 1965.
37. Newman, William L.; Buchinski, Edwin J. "Entry/Title Compression Code Access to Machine Readable Bibliographic Files." Journal of Library Automation 4:72-85, 1971.
38. Nugent, William R. "Compression Word Coding Techniques for Information Retrieval." Journal of Library Automation 1:250-260, 1968.
39. Payne, Charles. "University of Chicago Experimental Search Code." In: U.S. Library of Congress. Information Systems Office. The MARC II Format. Washington, Library of Congress, 129-131, 1968.
40. Plaister, Jean and others. "Conversion of a Regional Union Catalogue into Machine-Readable Form and Its Use as a Basis for Local Catalogue Conversion." LASER, 1973.
41. Ruecking, Frederick H. "Bibliographic Retrieval from Bibliographic Input; the Hypothesis and Construction of a Test." Journal of Library Automation 1:227-238, 1968.
42. Ruecking, Frederick H. "MARC at Rice University." In: Clinic on Library Applications of Data Processing, 1970, University of Illinois, Graduate School of Library Science, 66-77, 1971.
43. Scheiber, William D.; Thomas, George W. "An Algorithm for Compaction of Alphanumeric Data." Journal of Library Automation 4:198-206, 1971.
44. Schuegraf, E.J.; Heaps, H.S. "A Comparison of Algorithms for Data Base Compression by Use of Fragments as Language Elements." Information Storage and Retrieval 10:309-319, 1974.
45. Shannon, C.E. "A Mathematical Theory of Communication." Bell system Technical Journal 27:379-423,623-656, 1948.
46. Veneziano, Velma. "How to Develop the Record (Search) Key." 1971.
47. Williams, Martha E.; Shefner, Gordon J. "Data Element Statistics for the MARC II Data Base." Journal of Library Automation 9:89-100, 1976.